Proceedings

TWELFTH ANNUAL ALLERTON CONFERENCE ON CIRCUIT AND SYSTEM THEORY

October 2-4, 1974

Allerton House, Monticello, Illinois
Sponsored by the
Department of Electrical Engineering and the
Coordinated Science Laboratory of the
University of Illinois at Urbana-Champaign

Perkins

PROCEEDINGS

TWELFTH ANNUAL ALLERTON CONFERENCE
ON CIRCUIT AND SYSTEM THEORY

P. V. Kokotović
E. S. Davidson

Conference Co-Chairmen

Conference held

October 2, 3, 4, 1974
Allerton House
Monticello, Illinois

Sponsored by the

DEPARTMENT OF ELECTRICAL ENGINEERING
and the
COORDINATED SCIENCE LABORATORY
of the
UNIVERSITY OF ILLINOIS
at
Urbana-Champaign

FOREWORD

The Twelfth Annual Allerton Conference on Circuit and System Theory was held at the Allerton House on October 2-4, 1974. The Conference was cosponsored by the Department of Electrical Engineering and the Coordinated Science Laboratory of the University of Illinois at Urbana-Champaign. These PROCEEDINGS contain the papers presented at the Conference.

In addition to the large number of submitted papers, there were several organized sessions of invited papers. These sessions were: Three sessions on Optimization organized by D. Bertsekas in cooperation with L. McLinden and J. Liebman, two sessions on Large Scale Systems organized by D. D. Siljak and E. J. Davison, sessions on Information Retrieval organized by M. E. Williams, Threshold Logic organized by D. Hampel, Computer Architecture organized by E. S. Davidson, Bilinear Systems organized by D. L. Elliott, and Artificial Intelligence organized by D. L. Waltz. We are grateful to the organizers for their time and effort on behalf of the Conference. We thank the authors in these sessions for their prompt response and their excellent contributions. We are grateful to the session chairmen and all the authors for their effort and contribution to the success of the Allerton Conference.

Dr. H. H. Rosenbrock delivered the plenary address entitled "How Little Do We Know About Systems?" Dr. B. R. Myers served as toastmaster. F. P. Preparata was the speaker at the joint meeting with the IEEE Group on Information Theory, chaired by A. H. Haddad. We are much indebted to them.

The support and encouragement of Professor E. C. Jordan, Head of the Department of Electrical Engineering, Professor R. T. Chien, Director of the Coordinated Science Laboratory are appreciated and acknowledged. We thank our colleagues for their generous assistance in organizing the Conference: D. Bertsekas, J. B. Cruz, Jr., R. A. Flower, A. H. Haddad, W. Mayeda, G. Metze, W. R. Perkins, F. P. Preparata, M. B. Pursley, D. V. Sarwate, T. N. Trick, M. E. Van Valkenburg, D. L. Waltz, and N. Wax.

We are grateful to Mr. Harold Lawler for his handling of the business aspects of the Conference. We are also grateful to Mrs. Rosa Townsend for her help in the preparation of the Conference. Thanks are also due Mr. J. Patel for coordinating the projectionists and the visual aid equipment. Special thanks are due to Mrs. Rose Harris for her expert and skillful coordination of the Conference arrangements.

P. V. Kokotović
E. S. Davidson

TABLE OF CONTENTS

FOREWORD iii

PLENARY SESSION

Introduction: B. R. Myers, University of Maine

HOW LITTLE DO WE KNOW ABOUT SYSTEMS? 1
H. H. Rosenbrock, University of Manchester, England

I-A: OPTIMIZATION I
Organizer: D. P. Bertsekas
University of Illinois
Chairman: E. Polak,
University of California, Berkeley

A METHOD OF CONJUGATE SUBGRADIENTS FOR MINIMIZING NONDIFFERENTIABLE FUNCTIONS 8
Philip Wolfe, IBM, Yorktown Heights, New York

MINIMIZATION OF NONDIFFERENTIABLE FUNCTIONS WITH CONSTRAINTS 16
Claude Lemarechal, IRIA-LABORIA, France

MINIMIZING WELL-BEHAVED FUNCTIONS 25
Albert Feuer, Miami University, Oxford, Ohio

AN ALGORITHM FOR THE SOLUTION OF A NONDIFFERENTIABLE MINIMIZATION PROBLEM THAT AROSE IN GRAPH PARTITIONING STUDIES 35
Jane Cullum, W. E. Donath, P. Wolfe, IBM, Yorktown Heights, New York

ON GRADIENT PROJECTION 38
A. A. Goldstein, University of Washington

NONDIFFERENTIABLE OPTIMIZATION VIA APPROXIMATION 41
Dimitri P. Bertsekas, University of Illinois

I-B: STABILITY OF LARGE SCALE SYSTEMS
Organizer and Chairman: D. D. Siljak
University of Santa Clara

THE STABILIZABILITY OF LARGE INTERCONNECTED SYSTEMS ... 53
Edward J. Davison, University of Toronto

INTERCONNECTION OF UNSTABLE LINEAR SYSTEMS 63
C. A. Desoer and W. S. Chan, University of California, Berkeley

ON THE METHOD OF VECTOR LYAPUNOV FUNCTIONS 71
V. Lakshmikantham, University of Texas, Arlington

STABILITY ANALYSIS OF STOCHASTIC COMPOSITE SYSTEMS 77
Anthony N. Michel and Robert D. Rasmussen, Iowa State University

TOLERANCE OF DECENTRALLY OPTIMAL CONTROLLERS TO NONLINEARITY AND COUPLING 87
Stein Weissenberger, NASA Ames Research Center

I-C: SWITCHING AND AUTOMATA THEORY
Chairman: F. P. Preparata, University of Illinois

REMARKS ON ALGEBRAIC DECOMPOSITION OF PERMUTATION AUTOMATA 96
Vijay B. Aggarwal, University of Vermont and David E. Muller, University of Illinois

A NEW ALGORITHM FOR DETERMINING TIME ISOMORPHISM OF DIRECTED GRAPHS 106
Larry E. Druffel and Douglas C. Schmidt, Vanderbilt University

ON THE PERIODICITY OF AN ITERATION OF FINITE-VALUED FUNCTIONS 115
Pawel Kerntopf, Polish Academy of Sciences

DISCRETE SZEGÖ POLYNOMIALS AND AN O(NlogN) ALGORITHM FOR ERROR EVALUATION 120
F. Chin and K. Steiglitz, Princeton University

I-D: DISCRETE SYSTEMS AND NETWORKS
Chairman: K. A. Pullen, Jr., Ballistic Research Labs., Aberdeen Proving Ground, Md.

ON THE COMPLEXITY OF A CLASS OF CONCENTRATION SWITCHING NETWORKS 130
Gerald M. Masson, University of Pittsburgh

GRAPH DRAWING - A THEORETICAL FOUNDATION 137
G. Cottafava and O. D'Antona, Universita di Pavia, Italy

II-A: OPTIMIZATION II
Organizers: L. McLinden and D. P. Bertsekas University of Illinois
Chairman: L. McLinden

THE OPTIMAL CONTROL OF HAMILTONIAN SYSTEMS 143
Earl R. Barnes, IBM, Yorktown Heights, New York

UNCONSTRAINED METHODS IN OPTIMIZATION 153
O. L. Mangasarian, University of Wisconsin, Madison

APPROXIMATION TECHNIQUES IN GRADIENT PROJECTION ALGORITHMS 161
H. Mukai and E. Polak, University of California, Berkeley

II-B: ESTIMATION AND LQG PROBLEMS
Chairman: A. H. Haddad, University of Illinois

DISCRETE OPTIMAL LINEAR SMOOTHING FOR SYSTEMS WITH UNCERTAIN OBSERVATIONS 171
R. A. Monzingo, Hughes Aircraft Company

PRECISION ON-BOARD ATTITUDE REFERENCE USING THE GENERALIZED INVERSE 180
T. T. McElroy and R. P. Iewns, TRW Systems

A TIME DOMAIN ANALYSIS OF OPTIMAL FILTERING IN MULTIRATE SYSTEMS 185
Johnnie C. Liang and Michael G. Strintzis, University of Pittsburgh

MINIMAX TERMINAL STATE ESTIMATION UNDER EXPONENTIATED GENERALIZED QUADRATIC LOSS 190
C. Padilla, University of Illinois and
M. Mintz, University of Pennsylvania

AN EXTENSION OF RADNER'S THEOREM TO NONZERO-SUM GAMES 199
Tamer Basar, Marmara Research Institute, Turkey

A RECURSIVE ALGORITHM FOR IMAGE ENHANCEMENT 207
Rodolfo Feick and Michael G. Strintzis, University of Pittsburgh

ON AN INTEGRAL EQUATION IN PHASE ESTIMATION 212
N. C. Mohanty, State University of New York at Buffalo

II-C: DIGITAL SYSTEMS
Chairman: G. Metze, University of Illinois

TRANSITION PARTITIONS FOR STATE ASSIGNMENT IN SEQUENTIAL MACHINES 217
A. Jai and J. W. Gault, North Carolina State University

DESIGN EQUATION GENERATION FOR ASYNCHRONOUS SEQUENTIAL CIRCUITS USING PARTITION THEORY 224
Dwight H. Sawin, III, USAECOM, Ft. Monmouth and Gary K. Maki, University of Idaho

A DESIGN TECHNIQUE FOR TOTALLY-SELF-CHECKING ASYNCHRONOUS CIRCUITS 233
S. M. Reddy, University of Iowa

TOTALLY-SELF-CHECKING CHECKERS FOR A CLASS OF SEPARABLE CODES 238
M. J. Ashjaee and S. M. Reddy, University of Iowa

A NEW REPRESENTATION FOR MULTIPLE FAULTS IN COMBINATIONAL DIGITAL CIRCUITS 244
G. Metze, University of Illinois and C. Cha, IBM, Yorktown Heights, New York

HAZARD ELIMINATION IN TWO-VALUED CIRCUITS BY MULTIVALUED TECHNIQUES 254
Edgar DuCasse, Brooklyn College, CUNY

THE NUMBER OF HAZARD-FREE MULTIPLE-VALUED FUNCTIONS ... 263
Edgar DuCasse, Brooklyn College, CUNY

II-D: PATTERN RECOGNITION
Chairman: R. S. Michalski
University of Illinois

ON TESSELLATION AUTOMATA CAPABLE OF MIXED PARALLEL-SEQUENTIAL PROCESSING 268
William I. Grosky, Georgia Institute of Technology and Frank Tsui, IBM Corporation, Gaithersburg, Maryland

STOCHASTIC TREE SYSTEMS FOR SYNTACTIC PATTERN RECOGNITION 278
B. K. Bhargava, Regenstrief Institute, Indianapolis, Indiana and K. S. Fu, Purdue University

OPTIMAL FEATURES SELECTION ALGORITHM FOR A BINARY-PARTITIONED CLASSIFICATION SCHEME-ANALYSIS 288
Paul P. Wang and Richard C. Burns, Duke University

A CHARACTER PERCEPTION-RECOGNITION SYSTEM 298
B. E. Gala and D. A. Michalopoulos, California State University, Fullerton

A SIMPLE CORRELATION METHOD FOR HANDWRITTEN CHARACTER RECOGNITION 307
Michael E. Irrgang and Stephen S. Yau, Northwestern University

A SYSTEMS APPROACH TO IDEAL RETRIEVAL 317
Keats A. Pullen, Jr. and Charles W. Hoffman, US Army Ballistic Research Labs., Aberdeen Proving Ground, Maryland

II-E: CONTROL SYSTEMS
Chairman: W. R. Perkins, University of Illinois

DUAL CONTROL OF LINEAR DYNAMIC SYSTEMS WITH UNKNOWN PARAMETERS USING BAYESIAN PREPOSTERIOR ANALYSIS 324
R. K. Mehra, Harvard University

OBSERVERS AND PSEUDO OBSERVERS FOR LINEAR TIME DELAY SYSTEMS 333
R. V. Gressang, Air Force Flight Dynamics Laboratory, Wright-Patterson AFB, Ohio

ON THE ORDER DISCRIMINATION OF LINEAR DISCRETE-TIME SYSTEM MODELS USING THE INSTRUMENTAL VARIABLE APPROACH 344
Naresh N. Gupta, Miami University, Oxford, Ohio

DECOUPLING OF LINEAR SYSTEMS BY OUTPUT FEEDBACK 351
N. Viswanadham and D. P. Atherton, University of New Brunswick, Canada

A GRAPH MATRIX APPROACH TO STRUCTURAL CONTROLLABILITY 361
Jonas Butvila and Roland Priemer, University of Illinois at Chicago Circle

A TOPOLOGY FOR STATE CONSTRUCTION 369
Norman Y-K Foo, State University of New York at Binghamton

ON A SYSTEMATIC DESIGN OF NONLINEAR TIME-DELAY PROCESSES 379
M. Jamshidi, Pahlavi University, Shiraz, Iran

III-A: OPTIMIZATION III
Organizers: J. Liebman and D. P. Bertsekas
University of Illinois
Chairman: J. Liebman

A STUDY OF A HEURISTIC ALGORITHM 393
T. C. Hu and M. L. Lenard, University of Wisconsin, Madison

INTEGER AND MIXED-INTEGER PROGRAMMING MODELS: GENERAL PROPERTIES 396
R. R. Meyer, University of Wisconsin, Madison

AN EFFICIENT ALGORITHM FOR MAXIMUM K-MATCHING IN WEIGHTED GRAPHS 398
Lee J. White, Ohio State University

A NONLINEAR INTEGER PROGRAMMING ALGORITHM FOR THE DESIGN OF RADAR WAVEFORMS 411
Arthur I. Cohen, Systems Control, Inc.

SOLUTION TECHNIQUES FOR CERTAIN NONLINEAR MULT MULTICOMMODITY FLOW PROBLEMS 421
R. Klessig, Bell Telephone Laboratories, Holmdel, New Jersey

AN ALGORITHM FOR PLACING INTERCONNECTED OBJECTS IN A GIVEN CONFIGURATION AND ITS APPLICATIONS 431
R. R. Peiser, Schlumberger Well Services, Houston Texas, and S. C. Lee, University of Houston

III-B: STABILIZATION
Chairman: J. B. Cruz, Jr., University of Illinois

CONTROLLABILITY, POLE PLACEMENT AND STABILIZABILITY IN LARGE SCALE SYSTEMS 441
U. Ozguner and W. R. Perkins, University of Illinois

ON EXACT MODEL MATCHING 450
R. Rutman and Y. Shamash, Tel-Aviv University, Israel

THE NECESSARY STRUCTURE OF MULTIVARIABLE REGULATORS ... 457
B. A. Francis and W. M. Wonham, University of Toronto

STABILITY CRITERIA FOR TWO-VARIABLE POLYNOMIALS 459
D. D. Siljak, University of Santa Clara

ON THE STABILIZATION OF LINEAR SYSTEMS 469
Medhat M. El-Geneidy, Alexandria University, Egypt and Roger L. Berger, McDonnell Douglas Corporation, St. Louis, Missouri

ON THE STABILIZATION OF A CLASS OF NONLINEAR SYSTEMS VIA HYPERSTABILITY 479
Tudor Ionescu and R. V. Monopoli, University of Massachusetts

REGION OF STABILITY FOR NONLINEAR DIFFERENCE EQUATIONS 485
A. S. C. Sinha, Indiana Institute of Technology

SUSTAINED OSCILLATIONS FOR A CERTAIN CLASS OF n^th ORDER NONLINEAR SYSTEM 490
K. L. Dudkikar, Mitre Corporation, Virginia

III-C: INFORMATION RETRIEVAL
Organizer and Chairman: M. E. Williams
University of Illinois

AN OPERATIONAL EXPERIMENT FOR THE MARKETING OF SCIENTIFIC AND TECHNICAL INFORMATION 495
Donald J. Hillman, Lehigh University

THE USE OF HIERARCHIC CLUSTERING IN FORMING A SIMILARITY MEASURE 502
Scott E. Preece, Illinois Institute of Technology

AUTOMATIC CLASSIFICATION OF SCIENTIFIC LITERATURE USING CO-CITATION CLUSTERING 512
Henry G. Small, Institute for Scientific Information, Philadelphia and Belver C. Griffith, Drexel University, Philadelphia

THE EFFECT OF TERMINAL SPEED ON SEARCH PATTERNS IN INTERACTIVE INFORMATION RETRIEVAL 522
Roger K. Summit, Lockheed Missiles & Space Company, Inc.

III-D: THRESHOLD LOGIC
Organizer and Chairman: Dan Hampel
RCA Laboratories

LOGIC NETWORK SYNTHESIS USING DIGITAL-SUMMATION THRESHOLD-LOGIC GATES 525
S. L. Hurst, University of Bath, England

STATISTICAL ANALYSIS OF THRESHOLD LOGIC CIRCUITS 535
C. R. Baugh and B. A. Wooley, Bell Laboratories, Holmdel, New Jersey

M-OUT-OF-N DETECTION USING THRESHOLD LOGIC 542
B. A. Wooley and C. R. Baugh, Bell Laboratories, Holmdel, New Jersey

MULTI-THRESHOLD MULTI-VALUED LOGIC CIRCUITS 547
A. Druzeta, A. S. Sedra, Z. G. Vranesic, University of Toronto

MULTITHRESHOLD LOGIC AND BOOLEAN RING SYNTHESIS 557
Tich T. Dao, Signetics Corporation, Sunnyvale, California

WEIGHTED-MAJORITY-LOGIC DECODING USING NONORTHOGONAL PARITY CHECKS--AN APPLICATION 567
W. E. Robbins, North Carolina State University

IV-A: LARGE SCALE SYSTEMS
Organizer and Chairman: E. J. Davison
University of Toronto

RECENT RESULTS IN LARGE SCALE SYSTEMS 574
H. H. Rosenbrock, University of Manchester, England

AN INTEGRATED APPROACH TO LONG RANGE PLANNING FOR URBAN SYSTEMS 580
H. S. Rao, Systems Control, Inc., Palo Alto, California

COMPUTATIONAL METHODS FOR THE STEADY-STATE KALMAN FILTER IN HIGH ORDER NAVIGATIONAL MODELS 588
David L. Kleinman, University of Connecticut and Allan J. Kleinman, Dynamics Research Corporation, Wilmington, Massachusetts

LARGE SYSTEMS MODELING FOR COORDINATING ATMOSPHERIC PROCESSES AND ELECTRIC POWER OPERATION 599
James Gruhl, MIT

WHEN IS A COMPLEX ECOSYSTEM STABLE? 608
D. D. Siljak, University of Santa Clara

IV-B: COMPUTER ARCHITECTURE
Organizer and Chairman: E. S. Davidson
University of Illinois

CONCURRENCY IN GENERALIZED RADIX NON-RESTORING DIVISION 628
D. E. Atkins and Ü. Kalaycioglu, University of Michigan

CRITICAL-SET ALGORITHMS FOR PROGRAM LOCALITY IMPROVEMENT 641
Domenico Ferrari, University of California, Berkeley

A MODULAR SISD PROCESSOR 649
R. G. Cornell, Bell Laboratories, Naperville, Illinois

SCHEDULING OF MULTICONFIGURABLE PIPELINES 658
A. Thampy Thomas, Intersil, Inc. and
Edward S. Davidson, University of Illinois

SUGGESTIONS FOR IMPROVEMENTS TO A PIPELINED COMPUTER SYSTEM 670
Lawrence Cheung and Frederic J. Mowle, Purdue University

DYNAMIC MICROPROGRAMMING IN FORTRAN ENVIRONMENT 678
Philip S. Liu and Frederic J. Mowle, Purdue University

IV-C: NUMERICAL METHODS
Chairman: T. N. Trick, University of Illinois

NONSYMMETRIC PERMUTATIONS AND THE OPTIMAL BORDERED TRIANGULAR FORM OF A SPARSE MATRIX 686
A. Sangiovanni Vincentelli, Politecnico di Milano, Italy

NONLINEAR TRANSIENT ANALYSIS PROGRAMS: A COMPARISON .. 695
George W. Zobrist, University of South Florida

A GRADIENT METHOD FOR FINDING THE PERIODIC RESPONSE OF NONLINEAR SYSTEMS 703
M. S. Nakhla and F. H. Branin, Jr., University of Waterloo, Canada

A NEW TECHNIQUE FOR PARTIAL FRACTION EXPANSION WITH MULTIPLE POLES 705
Munetaka Konomi and Nobuyoshi Yoshida, Nippon Electric Company, Ltd., Japan

ABSOLUTELY STABLE INTEGRATION METHOD BASED ON THE LAPLACE TRANSFORM INVERSION 713
K. Singhal, J. Vlach, M. Nakhla, University of Waterloo, Canada

NUMERICAL LAPLACE INVERSION 720
George E. Cook and Francis M. Wells, Vanderbilt University

ACCURATE COMPUTATION OF THE TRANSIENT RESPONSE WITH TIGHT ERROR BOUNDS 730
R. M. Wadhwani and R. M. Kline, Washington University, St. Louis, Missouri and N. B. Rabbat, IBM Corporation, Hopewell Junction, New York

NON-LINEAR REALIZABILITY OF INPUT/OUTPUT TASKS: AN AUTOMATA-MOTIVATED SYNTHESIS APPROACH 739
Martin E. Kaliski, Northeastern University

IV-D: INFORMATION THEORY AND CODING
Chairman: D. Sarwate, University of Illinois

BLOCK CODING FOR STATIONARY GAUSSIAN SOURCES WITH MEMORY UNDER A SQUARE-ERROR FIDELITY CRITERION 740
Harry H. Tan, Princeton University

OPTIMALITY OF DELTA MODULATOR 749
G. Thomas and V. V. S. Sarma, Indian Institute of Science, Bangalore, India

VARIABLE-LENGTH ENCODING FOR DISCRETE MEMORYLESS COMPOSITE SOURCES 759
P. S. Budihardjo and J. W. Mark, University of Waterloo, Canada

EVALUATING PERFORMANCE OF CODES FOR SPREAD SPECTRUM MULTIPLE ACCESS COMMUNICATIONS 765
Michael B. Pursley, University of Illinois

SIGNAL DESIGN FOR ASYNCHRONOUS MULTIPLE ACCESS COMMUNICATIONS 775
N. C. Mohanty, State University of New York at Buffalo

COMBINATORIAL DECODING OF CERTAIN CODES 782
M. Rahman and I. F. Blake, University of Waterloo, Canada

PROPERTIES OF GENERALIZED PLESS CODES 787
Ian F. Blake, University of Waterloo, Canada

IV-E: SINGULAR PERTURBATION THEORY
Chairman: N. Wax, University of Illinois

A DIAGONALIZATION TECHNIQUE FOR SINGULAR PERTURBATION PROBLEMS 790
K. W. Chang, University of Calgary, Canada

NUMERICAL INTEGRATION OF GENERAL STIFF INITIAL-VALUE PROBLEMS BY SINGULAR PERTURBATION METHODS 795
Leon Lapidus, Princeton University and
Richard C. Aiken, Eidgenössiche Technische Hochschule, Zürich, Switzerland

FURTHER RESULTS ON SINGULAR PERTURBATIONS AND SINGULAR ARCS 803
R. E. O'Malley, Jr., University of Arizona and
Antony Jameson, Courant Institute, New York

APPROXIMATION OF HIGH ORDER LINEAR SYSTEMS BY LOWER ORDER MODELS 809
J. M. Siret and G. Michailesco, Laboratoire de Signaux et Systemes, France

V-A: BILINEAR SYSTEMS
Organizer and Chairman: D. L. Elliott
Washington University
St. Louis

BILINEAR SYSTEMS: GENESIS AND DEVELOPMENT 814
R. R. Mohler, Oregon State University

OPTIMAL CONTROL ON LIE GROUPS 823
John Baillieul, Harvard University

LINEARIZATION AND BILINEARIZATION OF CONTROL SYSTEMS 834
Arthur J. Krener, University of California, Davis

CONTROLLABILITY AND ACCESSIBILITY CRITERIA FOR BILINEAR SYSTEMS 844
David L. Elliott, Washington University, St. Louis

ANALYSIS OF BILINEAR STOCHASTIC SYSTEMS 849
Alan S. Willsky and Steven I. Marcus, MIT and
David N. Martin, Raytheon Company

V-B: ARTIFICIAL INTELLIGENCE
Organizer and Chairman: D. L. Waltz,
University of Illinois

TRACKING RANDOMLY MOVING OBJECTS WITH A COMPUTER VISION SYSTEM 858
Vincent C. Jones, University of Illinois

FEATURE EXTRACTION FROM COLOR CONTRAST IN VISUAL SCENES 862
Stephen A. Underwood and J. K. Aggarwal, University of Texas at Austin

AUTOMATIC VISUAL INSPECTION OF HYBRID CIRCUITS 863
Wesley E. Snyder, University of Illinois

A CONSTRAINT SATISFACTION METHOD FOR INFERENCE MAKING 866
John Gaschnig, Carnegie-Mellon University

NATURAL LANGUAGE BASED INFORMATION RETRIEVAL 875
R. P. Gabriel and D. L. Waltz, University of Illinois

VARIABLE-VALUED LOGIC: RECENT DEVELOPMENTS 885
R. S. Michalski, University of Illinois

V-C: OPTIMIZATION AND OPTIMAL CONTROL
Chairman: D. A. Hanson, Ohio State University

A CONSUMER THEORY FOR DURABLE GOODS; THE AUTOMOBILE MARKET AND WELFARE IMPLICATIONS 887
Donald A. Hanson, Ohio State University

A DYNAMIC MODEL OF EXCHANGE RATE IN RELATION TO INTERNATIONAL TRADING 897
K. G. Sun and O. D. Osborne, Oregon State University

A NEGATIVE-POSITIVE BARRIER METHOD FOR NONLINEAR PROGRAMMING 902
Christakis Charlambous, University of Waterloo, Canada

COMPUTATIONAL MERITS OF EXTRAPOLATION IN LEAST pth APPROXIMATION AND NONLINEAR PROGRAMMING 912
J. W. Bandler and W. Y. Chu, McMaster University, Canada

THE TOLERANCE-TUNING PROBLEM: A NONLINEAR PROGRAMMING APPROACH 922
J. W. Bandler and P. C. Liu, McMaster University, Canada

OPTIMAL CONTROL WITH TRANSITORILY ENLARGED CONTROL CONSTRAINT SETS 932
E. D. Eyman and M. B. Subrahmanyam, University of Iowa

RENDEZVOUS OF DISCRETE SYSTEMS IN A TARGET AREA 940
Dong Hak Chyung, University of Iowa

V-D: CIRCUIT THEORY AND DESIGN
Chairman: M. E. Van Valkenburg
University of Illinois

OPTIMAL CIRCUIT DESIGN WITH A SENSITIVITY PENALTY 942
T. H. Chau and R. Saeks, Texas Tech University

DECOMPOSITION THEOREM - SYNTHESIS OF INHOMOGENEOUS LADDER NETWORKS 948
Ting N. Lee, George Washington University and David P. Brown, University of Wisconsin, Madison

SOME PROPERTIES AND APPLICATIONS OF MATRIX CONTINUED FRACTION 957
L. S. Shieh and F. F. Gaudiano, University of Houston

SENSITIVITY INVARIANCE USING STATE VARIABLES 967
M. Lal, Wichita State University and J. S. Sohal and H. Singh, University of Roorkee, India

UNIDIRECTIONAL RLC TRANSMISSION LINES 975
V. Burke and D. Hazony, Case Western Reserve University, Cleveland, Ohio and R. J. Duffin, Carnegie-Mellon University

ADDITIONAL TYPES OF MUTATORS AND ACTIVE RC SYNTHESIS USING MUTATORS 982
Tadao Murata, University of Illinois at Chicago Circle

DESIGN OF ACTIVE RC NETWORKS BY NONLINEAR PROGRAMMING 991
Nicholas Kyriakopoulos and Gilbert Garduno, George Washington University

DESIGN EQUATIONS OF A HYSTERETIC DC-TO-DC CONVERTER ... 997
A. S. Kislovski, Bern, Switzerland

THE WIEN BRIDGE OSCILLATOR REVISITED (OR, THE WIEN BRIDGE OSCILLATOR USED AS A KEYED OSCILLATOR OR A BANDPASS FILTER) 1003
Richard E. Held, Harris Corporation, Melbourne, Florida

V-E: DETECTION AND ESTIMATION
Chairman: M. B. Pursley, University of Illinois

DEPENDENT SAMPLING IN NON-ADAPTIVE NONPARAMETRIC DETECTORS 1007
Saleem A. Kassam and John B. Thomas, Princeton University

THE OPTIMIZATION AND PERFORMANCE OF DETECTORS BASED ON PARTITION TESTS 1016
W. J. Bushnell, Bell Laboratories, Whippany, New Jersey and L. Kurz, Polytechnic Institute of New York

THE EFFECT OF INTERSYMBOL INTERFERENCE ON THE PERFORMANCE OF FM DIGITAL SYSTEMS 1024
P. Papantoni-Kazakos, Rice University and I. Paz, NASA, Houston

SOME RESULTS ON THE CHARACTERIZATION OF RANDOM SEQUENCES AND RELATED PULSE TRAINS 1029
G. L. Wise and J. B. Thomas, Princeton University

ESTIMATION OF PARAMETERS IN SYSTEMS OF RELATIVELY SLOW TIME VARIATION 1039
Philip H. Fiske and William L. Root, University of Michigan

MINIMAX PARAMETER ESTIMATION UNDER GENERALIZED QUADRATIC LOSS WITH A COMPACT PARAMETER SPACE 1048
R. Padilla, University of Illinois and M. Mintz, University of Pennsylvania

HOW LITTLE DO WE KNOW ABOUT SYSTEMS?

H.H. ROSENBROCK
UMIST, MANCHESTER, England.

The rapid development during recent years of our knowledge of dynamical systems may give the impression that little remains to be discovered. This is far from true, and the talk will describe some of the areas in which our present knowledge is inadequate. The viewpoint will be that of an engineer whose interest is in understanding and controlling large-scale industrial plants.

HOW LITTLE DO WE KNOW ABOUT SYSTEMS?

My first visit to the United States was in 1958, and I took the opportunity to meet one of the senior and most respected men working on automatic control. Control theory, he assured me, was worked out, and it had become impossible to find suitable new topics for Ph.D. students. The time had come to move on to other things.

I mention this because I have detected in the last three years a similar attitude to the subject. Of course, in 1958 the great expansion in state-space theory, optimisation, and aerospace problems was hidden in the future. Not very far in the future: the Rand Report by Bellman, Glicksberg and Gross appeared in that year, and the First IFAC Congress was only two years away.

The new impetus which was given to the subject around that time seems to me to have been largely spent by 1970. Details no doubt remained to be added, and numerical methods are still not so good as we might wish, but optimal control theory will probably not change very much from the form which it had taken by 1970. Moreover, the aerospace problems which motivated much of the work had been decisively solved, and support from that industry began to be reduced.

In 1958, what my informant was really saying was that one branch of control theory had been taken as far as was possible at that time - the theory of single-loop systems treated by frequency-response methods using pencil and paper. Similarly, I believe that present disenchantment springs from the exhaustion of one specific line of development. To equate this with a complete understanding of systems and their control is much too optimistic, as I shall hope to show.

THEORY AND PRACTICE

Let me begin by commenting on the relation between theory and practice in engineering, which seems to me to be often misunderstood. First, it is very rare indeed for a purely theoretical advance to lead directly to an advance in engineering practice. Perhaps the Brequet watch spring was an example, but one has great difficulty in finding others. Usually engineer-advances are based on newly-discovered physical facts (electrical generators, transistors) or an intuitive and empirical approach (Watt's governor, the aircraft gas-turbine).

Secondly, engineering situations are always much more complicated than theory assumes. Indeed, one may say that the assumptions of any theory which the engineer uses are never true of the situation to which he applies it. It is one of the engineer's chief responsibilities to judge whether such discrepancies will invalidate his design.

Thirdly, it is no more than 20 or 30 years since advanced mathematical theory began to be applied to engineering problems. This, in passing, poses a problem to engineers of my generation who have to make themselves familiar with new branches of mathematics. Perhaps it also poses a problem for younger engineers, whose training may suggest to them that engineering is more amenable to exact analysis than is really true.

As compared with this short history, the application of mathematics to the physical sciences has a history of at least three hundred years in the modern world, and a still longer history in antiquity. There the criteria are well understood: mathematical theory is valid if it explains known facts (Newton) or better still predicts new ones (Adams-Leverrier, Einstein) which can be experimentally verified.

In engineering science, however, we are not concerned with explaining the externally-given world. Rather we are concerned with what new things men might be able to do, or what old things they might do better. The scope is therefore limited only by human ingenuity, and there is a corresponding danger that theory may become a self-perpetuating and self-justifying activity. We need to develop criteria of validity in engineering science which will prevent such an aberrant proliferation, and it seems to me that there are two kinds of role that theory can play.

Engineering advances, as I have said, are usually made without a full theoretical analysis. In process control, for example, there was a highly successful art of applying controllers long before any coherent theory was available. Engineers with long experience could apply and tune pneumatic controllers and achieve excellent results. They could not explain clearly how they did this, and it took many years to acquire the skill.

Once the control theory which was developed during the second world war was made available, this art became amenable to analysis. Long experience was no longer essential, because the facts were explained in a clear, logical way. Because he had a theoretical framework, the engineer could understand better how existing devices worked. He was also placed in a position from which further advances became possible. This first role of theory, in providing understanding and preparing the way for new advances, is probably the most vital. The advances which follow are still likely to arise in a practical setting, but they would not be possible without the previous understanding.

The second role is simpler, and consists in the development of new methods of analysis, synthesis or design. These can be directly applied in practice and contribute to the improvement (usually in detail) of the engineering art. The methods of Nyquist, Bode, and Evans, besides their contribution to understanding, are outstanding early examples of such design methods.

Unfortunately much work which would be very defensible as a contribution to understanding is presented as a direct contribution to practice. The disillusion of practising engineers, to which this has led, is probably one of the chief reasons for the 'gap' between theory and practice. Design methods can only usefully be developed by those who are close to engineering applications, and we should probably do well to distinguish more clearly between the two roles of theory.

CONTROL IN INDUSTRY

If one regards the present state of control practice in industry, it is remarkable how much of it is based on empiricism and how little on theory. Oil refineries, chemical plants, aircraft gas-turbines, steelworks and most other industrial plants are controlled in a highly successful way. Yet the amount of control theory used in their design is usually quite small. It is true that theory provides an essential mental framework, together with knowledge of the technology of the process. This, however, is used chiefly to guide an intuitive or pragmatic approach to the design. The amount of calculation used in the design of the control system is generally small.

There are several reasons for this. Perhaps the chief one is that existing practice usually provides satisfactory control, so that the incentive to spend more effort is small. A second reason is that to obtain a valid mathematical description of the plant may require much effort, particularly at the design stage, before the plant is built.

These are practical reasons, but there are deeper and more theoretical difficulties as well. A detailed plant model, for example, requires much effort to develop. Yet we know that in many cases the information we require to control the plant is fairly simple. For example, the early frequency-response methods show that often the response to sinusoidal inputs, over a restricted frequency range, and known only approximately, is quite enough for design purposes. At present we lack methods for obtaining this rather simple and approximate knowledge without undertaking a detailed analysis, much of which will not be utilised.

Again, we have very little knowledge of the structural properties of large systems. Let me illustrate by a hypothetical example. Suppose that an engineer were to present me with a complete set of equations describing a chemical plant, and were to ask me how to control it. There is very little of practical value that I could tell him. But if he were to add "this is a distillation column, this is a reactor, ---- " and so on, I should be able to say much more. From physical knowledge we can exploit the structure which we cannot detect or analyse mathematically.

SYSTEMS THEORY

Indeed this question of structure seems to be central, yet it has been very little studied in systems theory. For this reason, branches such as electrical network theory have never been fully brought into the general stream of systems theory. We know, for example, many properties of an electrical network which follow from the fact that it consists of defined elements, connected in defined ways. These results ought to be special cases of more general theorems about systems, but we have very little in this way.

To take an example which has interested me, there is often an apparent redundancy in system descriptions. In a distillation column, we may write equations for the compositions (the mol fractions of each component) on each plate. Being mol fractions, these sum to unity. We could eliminate one component, but this is likely to give unacceptable errors when we recover the amount of this component by difference. On the other hand, if we attempt to integrate all the equations, then numerical errors will cause the sum of compositions to depart from unity. The redundancy in this example cannot be fully explained by uncontrollability or unobservability, which are the only available types of redundancy for systems in state-space form. Consequently we have to use a more general description of the system if we are to investigate it fully.

The same is true of linear electrical networks. If the equations for these are written in state-space form, we cannot easily relate their properties to those of their elements. Some of the system properties will in fact have been lost when the equations are brought to state-space form. If such phenomena can occur in linear systems, we may well expect similar or more complicated phenomena in nonlinear systems.

As a second example we may mention the problem of plant identification. Structural properties are central to this procedure, as has recently become quite clear, yet we still know very little about them.

ENGINEERING DESIGN

One of the chief justifications for systems theory is the contribution which it can make to engineering through the development of better design methods. Those who are not themselves in close touch with industrial practice may have the impression that there has already been a large contribution in this way. It is important to emphasise that this is not so.

In the design of process plants at the present day (chemicals, oil, food, steelmaking, etc.) it is quite exceptional for any significant effort to be spent on the dynamical behaviour of the control system. Attention is concentrated almost entirely on steady-state considerations and on the results of past experience. In the control of aircraft gas turbines the long-standing methods of frequency response may be used, but most of the design work proceeds empirically using a simulation. The same is true in most other industrial areas. There are indeed some exceptions, but by volume and by the value of the plant concerned they are in a small minority.

As I have already said, there are a number of obvious reasons for this situation. Existing control systems often perform quite well, and the scope for improvement is not outstandingly great. To obtain an improvement by existing methods often calls for long and expensive investigations. Moreover there are practical considerations of simplicity, reliability, ease of maintenance, etc., which present methods cannot directly take into account. All of these reasons lead to a situation where technical effort spent on the control system may give smaller returns than the same effort spent on modifications of the plant design.

There are two developments which I believe could improve this situation. The first is simply an increased understanding of the structure of systems and the influence which this has upon control. Most present methods used to produce control systems do not exploit any knowledge of the structure of the system. Optimal control, for example, attempts to solve the control problem without demanding of the designer (or offering to him) any knowledge of structure. Because of this structure, which can be exploited by the empirical practitioner, there may well be simpler and better solutions.

The second development is the ease with which it is now becoming possible to communicate with digital computers. Computers are admirable devices for computation, but they are very inefficient at other tasks. Present to a designer a tentative design for a system, and he will often be able to say at once "that is no good because ----- ". The defect stands out and is recognised almost instantaneously. To obtain this information from a computer demands a tedious process of checking all possible deficiencies, and it is often forgotten that the designer has to write out the list of deficiencies to be checked. This is more burdensome than recognising a few defects in a tentative solution.

My belief is that the interactive use of computers offers a much more powerful tool to designers than we have ever had before. Large algorithmic calculations will still be needed in some areas, but much of design is probably best done in an iterative manner, with the designer modifying and guiding the course of the design. For this purpose, however, we need new methods. They need to give to the designer a quick understanding of the results of his actions, and to allow him to suggest profitable improvements. Almost inevitably they must present this information to him graphically.

CONCLUSION

To summarise, I suggest that we stand today at the end of one period of development of our subject. It was a period motivated largely by one problem - the aerospace problem. It was also a period constrained by the available equipment to off-line algorithmic methods. These are difficult to apply to industrial control, and often give unnecessarily complicated answers. They do not demand nor provide any understanding of the structure of a system, and are therefore incapable of exploiting this structure.

We also stand, I believe, at the beginning of a period where understanding of structure will be a primary aim. We shall become increasingly able to make qualitative statements about the problems of control. We have new opportunities for using the best features of computers in collaboration with the quite different features of the human mind. In this way we may be able to tap more effectively the reserves of creative skill which are so under-used in our civilisation.

I, at least, look forward to such a development with hope and confidence.

A METHOD OF CONJUGATE SUBGRADIENTS FOR MINIMIZING NONDIFFERENTIABLE FUNCTIONS

PHILIP WOLFE
IBM Research Center
Yorktown Heights, New York

ABSTRACT

Some problems involving the minimization of nondifferentiable functions are described and needed notions from convex analysis cited. An example shows that the method of steepest descent will not solve such problems, but leads to a method--an extension of the method of conjugate gradients--which does.

SOME NONDIFFERENTIABLE PROBLEMS

Many problems of optimization require the minimization of a nondifferentiable function f of several variables $x = (x_1,..,x_n)$. Danskin [1] initiated the systematic study of an important class of these related to zero-sum games: letting $g(x,y)$ be the payment from a player using a strategy $x \in X$ to his opponent, who answers x with the strategy $y \in Y$, the first player will seek the minimum for $x \in X$ of

$$f(x) = \text{Max}\left\{g(x,y) : y \in Y\right\} . \tag{1}$$

No matter how smooth g is, f is almost always significantly nondifferentiable. In the simplest, but representative, case Y is finite, and we may write $f(x) = \text{Max}\,\{g_i(x) : i = 1,..,m\}$, and suppose each g_i to be differentiable. In Figure 1, $m=2$; f, like all the function we deal with, is differentiable almost everywhere, but that gives no comfort; differentiability fails just where we want to be, at the minimum.

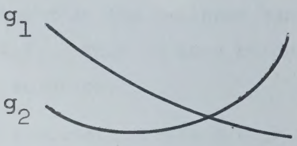


Fig. 1

If g above is convex in x for each y, then f is a convex function. Commonly g is even linear in x; and an important fact about any convex f is that there always is g, linear in x, so that (1) holds.

Problems of best L_∞ (or "Chebycheff") approximation form an interesting subclass of those above. In a simple case, a function $h(y)$ is to be approximated by a linear combination $\Sigma_{j=1}^{n} x_j f_j(y)$ of functions f_j at a finite number of points $y_1,..,y_m$, and the function to be minimized is $f(x) =$

$= \mathrm{Max}_i |h(y_i) - \Sigma_j \, x_j f_j(y_i)|$, or the maximum of the 2m functions $\pm[h(y_i) - \Sigma_j x_j f_j(y_i)]$. As the linear programming problem

$$\mathrm{Min}\,\{ z : z \geq \pm[h(y_i) - \Sigma_j x_j f_j(y_i)], \text{ all } i \} ,$$

it is best solved by the simplex method unless m or n is an extraordinary number; but for nonlinear approximation, for example if $x_j f_j(y)$ is replaced by $\exp[x_j f_j(y)]$, the problem of finding a good method is open, and being busily pursued.

One might question whether the difficulty of the previous problem is not due to rather arbitrary insistence on the L_∞ norm as a measure of goodness of fit; but there are many particular problems in which, as in the general minimax problem, nondifferentiability is inherent. We mention just two of especial interest: Fletcher's exact penalty function [2] and the eigenvalue problem whose solution will be given by Cullum [3] at this Conference.

Finally we describe a large--and, hopefully, important--class of problem coming from linear programming: problems which can in principle be solved by the simplex method, but whose great size and special structure raise the possibility that the reformulations sketched below may provide good computational methods.

The cutting-stock problem [4], the production-scheduling problem [5], and approximate duals to some difficult combinatorial problems such as the traveling salesman problem [6] and integer programming problems [7] can be posed as problems requiring the minimization of the convex, piecewise linear

$$f(x) = \mathrm{Max}\,\{ a_i + \langle u_i , x \rangle : i = 1,..,m \} , \qquad (2)$$

where x is of reasonable dimension but m is astronomical (eg. $100!$). We can deal with such a problem only if it has the feature that its data can be generated: basically that, given x , there is some procedure for determining a_i and u_i for which the Max of (2) is assumed; we do not require an explicit listing of all the data of the problem. Such is the case for the problems mentioned, as well as for the decomposable problems of the next paragraph.

Some very large linear programming problems can be posed as requiring the minimization of the linear function $\langle c, x \rangle + \Sigma_{k=1}^{K} \langle c_k , y_k \rangle$ under constraints of the form $A_k x + B_k y_k \leq d_k$ $(k=1,..,K)$, where the vectors c , c_k and the matrices A_k , B_k are of tractable size. Given x , the minimal values of $\langle c_k , y_k \rangle$ may be found by solving K independent linear programming problems; we say the problem has been decomposed. Since each of those values is a convex function of x , the problem may be stated as that of minimizing

$$f(x) \;=\; \langle c,x\rangle + \Sigma_k \; \mathrm{Min}\left\{\langle c_k, y_k\rangle : B_k y_k \le d_k - A_k x\right\} \quad , \qquad (3)$$

or equivalently (using linear programming duality)

$$f(x) \;=\; \langle c,x\rangle + \Sigma_k \; \mathrm{Max}\left\{\langle u_k, A_k x - d_k\rangle : u_k B_k + c_k \ge 0\right\} , \qquad (4)$$

showing f to be the maximum of the vast number of linear functions given by each K-tuple $(u_1,..,u_K)$ of basic solutions of the systems $u_k B_k + c_k \ge 0$ for $k=1,..,K$.

$$\begin{array}{|c|c|c|l}
\hline
A_1 & A_2 & A_3 & \ldots \quad \le d \\
\hline
B_1 & 0 & 0 & \ldots \quad \le d_1 \\
\hline
0 & B_2 & 0 & \qquad \le d_2 \\
\hline
0 & 0 & B_3 & .. \qquad \vdots
\end{array}$$

Fig. 2

The constraint structure of Figure --the transpose of that of the problem above--is in fact more common. The transformation of the preceding paragraph applies to its dual, yielding a "Lagrangian decomposition" in which the variables x are Lagrange multipliers for the joint constraints of the problem. The original decomposition algorithm [8] can be viewed as one way of determining the optimal multipliers; in exploring other ways of minimizing f , we seek to improve on that scheme. There is no shortage of proposals for improvement [9,10,11 and many others], and some computational evidence of its possibility [12], but at this time only the original algorithm has actually been used for general large-scale linear optimization.

NONDIFFERENTIABLE CALCULUS

We briskly review the mathematical background for our problems as given by Rockafellar [13] for convex functions. For simplicity, we suppose f finite-valued over all R^n .

For any x , the directional derivative

$$f'(x;d) \;=\; \lim_{t \to 0^+} \; [f(x+td) - f(x)]/t \qquad (5)$$

exists and is finite. The set

$$\partial f(x) \;=\; \{u : f(y)-f(x) \ge \langle u, y-x\rangle \;\; \text{for all} \;\; y\} , \qquad (6)$$

called the subdifferential of f at x , consisting of all subgradients of f at x , is compact and convex, and

$$f'(x;d) \;=\; \mathrm{Max}\;\{\langle u,d\rangle : u \in \partial f(x)\} \; . \qquad (7)$$

The point x minimizes f if and only if

$$f'(x;d) \geq 0 \text{ for all } d \iff 0 \in \partial f(x) \quad . \tag{8}$$

Danskin [1] anticipated (7) for minimax problems in which $g_x(x,y)$ is jointly continuous in (x,y): while the characterization (6) cannot be used when f is not convex, (7) holds with $\partial f(x)$ taken to be the convex hull of the set of all gradients $g_x(x,y)$ for which y attains the maximum in (1). In such a case, of course, (8) characterizes a local, not necessarily global, minimum of f : in fact, a stationary point. In this meeting Feuer [14] will describe a further weakening of the requirements on f still permitting the same general ideas to be used.

STEEPEST DESCENT

Demjanov [15] pointed out that $f'(x;d)$, for fixed x , is either nonnegative for all d (so that x is stationary) or has a unique minimizer $d : |d| = 1$, which is then the direction of steepest descent. (As an alternative definition, that d is also the limit as $r \to 0$ of $(x_r - x)/r$, where x_r is the minimizer of f in a sphere of radius r about x . See Figure 3.) He showed that such $d = - \text{Nr}\, \partial f(x)$, where "Nr S" denotes the point of smallest Euclidean norm in the convex hull of the set S. Nr S can be efficiently calculated when S is finite [16].

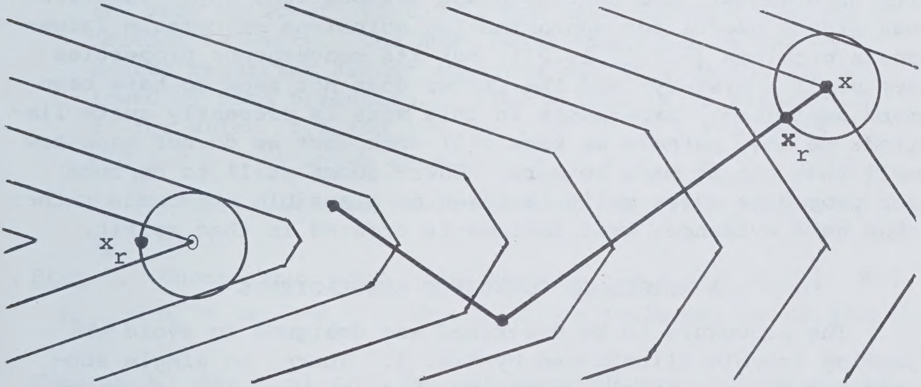


Fig. 3: Steepest descent

The method of steepest descent consists in the iteration of the step $x \leftarrow x+td$, where $d = -\text{Nr}\, \partial f(x)$ and t is chosen to minimize $f(x+td)$ for $t \geq 0$. While it works (although not quickly) for differentiable functions, in that a limit point of the iterates is a stationary point of f , it does not work in general. Fig. 3 sketches contours of the convex function $f(x,y) = \text{Max} \{3x \pm 2y\ ,\ 2x \pm 10y\}$ and two legs of a steepest descent path. One finds that the generated points converge to (0,0), which is certainly not stationary. This phenomenon, called zigzagging or jamming, can also happen for smooth functions in the presence of constraints [17]. Even when a problem is such that it cannot occur, as for smooth functions without constraints, the problem will be hard if its gradient can fluctuate widely around the solution.

REQUIREMENTS OF AN ALGORITHM

Bertsekas and Mitter review most of the literature on general methods for nondifferentiable optimization prior to 1973 in the valuable Introduction to their paper [18]. The proposals that have been made vary in the amount of information that must be developed about f at or near a given point. We have been most interested in problems, like (2,3,4) above, which suffer the strictest requirements: given x we can calculate $f(x)$ and one member of $\partial f(x)$, but cannot (conveniently) find any other members. Such restriction may seem strange to one used to the minimax problems of the first Section, but it held for all the problems we experimented on in a recent study [19], and can be seen to hold pretty firmly for decomposable problems.

Only a few previous methods are aimed at operating under such restriction, notably "subgradient optimization" [19, and its References], and cutting-plane methods [eg. 20]. The first has proved useful for approximating solutions of certain large-scale problems [6,7,12,19,21], but its convergence properties are still a mystery; and the latter does not seem to have been much exploited. Experience in this area is presently quite limited: we have methods we know will work, but we do not know how well they can be made to work. There seems still to be room for proposals whose merit is based on plausible arguments rather than hard evidence; what follows is offered in that spirit.

A METHOD OF CONJUGATE SUBGRADIENTS

The procedure to be described was designed to avoid the jamming trouble illustrated by Fig. 3. There, no single subgradient gives a usable direction; but, knowing [13, Theorem 24.6] that $\partial f(x)$ is approximated by the set of all subgradients arising in any neighborhood of x, in the course of generating successive points x_k we will form the bundle $G_k = \{..,g_{k-1},g_k\}$ of correspoinding subgradients $g_k \in \partial f(x_k)$. The direction of motion d_k from x_k will be formed as if steepest descent were appropriate: $d_k = -\mathrm{Nr}\, G_k$. Then a suitable search along the ray $\{x_k+td_k : t \geq 0\}$ will find x_{k+1}, an approximate minimizer of f on the ray, and $g_{k+1} \in \partial f(x_{k+1})$, completing the basic iterative step. Applied to our example, on the second step the bundle is $(3,2),(3,-2)$, and $d = (-3,0)$; we are finished, since $f(x+td) \rightarrow -\infty$ as $t \rightarrow \infty$, for any x.

The complete algorithm, motivation, and proofs of the convergence results are given elsewhere [22]. Below we give the heart of the matter, the "Main algorithm", which approximately minimizes f in the sense of Theorem 1 below.

Given: $x_o \in R^n$, $g_o \in \partial f(x_o)$, and positive ε, δ, b, $m_2 < m_1 < \frac{1}{2}$. Set $G_o = \{g_o\}$, $a_o = 0$. For $k = 0,1,\ldots$:

Step 1. Set $d_k = -\mathrm{Nr}\, G_k$. If $|d_k| \leq \varepsilon$, go to Step 2; otherwise, to Step 4.

Step 2. If $a_k \leq \delta$, the algorithm is done; else do Step 3.

Step 3. Set $x_{k+1} = x_k$, $g_{k+1} = g_k$, $G_{k+1} = \{g_k\}$, $a_{k+1} = 0$. Cycle k is finished; go to Step 1.

Step 4. Define $g(t)$ as the subgradient that can be obtained in $\partial f(x_k + td_k)$, for any $t > 0$. Define the intervals

$$L = \{t : t > 0,\ f(x_k+td_k) - f(x_k) \leq -m_1 |d_k|^2 t\},$$

$$R = \{t : t > 0,\ \langle g(t), d_k\rangle \geq -m_2 |d_k|^2\}.$$

(L and R cover $\{t : t > 0\}$, and have positive overlap if neither is empty. The next requirement can be met by a search requiring either stepping out or stepping in, followed by some bisection.) Find either (a) a sequence of t for which $f(x_k+td_k) \rightarrow -\infty$, or (b) $t \in R-L$ with $t|d_k| < b$, in which case set $x_{k+1} = x_k$, $g_{k+1} = g(t)$, or (c) $t \in L \cap R$, in which case set $x_{k+1} = x_k + td_k$ and $g_k = g(t)$.

Step 5. Choose any $H \subseteq G_k$ and set $G_{k+1} = H \cup \{-d_k, g_{k+1}\}$ and $a_{k+1} = a_k + |x_{k+1} - x_k|$. Cycle k is finished; go to Step 1.

Theorem 1. Let $|g| < C$ for all subgradients g found. The main algorithm generates either a sequence x_k for which $f(x_k) \rightarrow -\infty$, or a point x_k for which $f(x_k + td_k) \rightarrow -\infty$ as $t \rightarrow \infty$, or terminates at step $k \leq 4C^2[f(x_0) - f(x_k)]/m_2 \delta \varepsilon^3$ in a point x_k such that, for all $y \in R^n$,

$$f(y) - f(x_k) \geq -\varepsilon|y - x_k| - 2C(b+\delta).$$

In the complete procedure, b is chosen quite small, as m_2 may be, while ε, δ are initially sizable. The main algorithm is run repeatedly, decreasing ε, δ for each run.

While the algorithm is, of course, aimed at the nondifferentiable problem, we were greatly encouraged by its unexpected relationship to a method of basic importance for smooth problems--that of conjugate gradients, In that respect and others it has much in common with the methods of Lemarechal [23,24,25], who points out that this method and his are extensions of the variable metric method of Davidon [26, and its References].

Theorem 2. Let f be quadratic. Set $\varepsilon = \delta = 0$ in the main algorithm and replace Step 4 by a procedure which finds, if it exists, the exact minimum of f on $\{x_k + td_k : t>0\}$. For some k no greater than the number of distinct positive eigenvalues of the quadratic form for f , either $\nabla f(x_k) = 0$ or $f(x_k + td_k) \rightarrow -\infty$ as $t \rightarrow \infty$. The sequences x_k, g_k, d_k coincide with those generated by the method of conjugate gradients [27, Section 9, Modification 1] (reading x_i, r_i, p_i for the quantities above).

REFERENCES

1. J.M. Danskin, The Theory of Max-Min. Springer-Verlag 1967.

2. R. Fletcher, An ideal penalty function for constrained optimization. A.E.R.E. Harwell, December 1973.

3. J. Cullum, W.E. Donath, and P. Wolfe, An algorithm for minimizing certain nondifferentiable convex functions. RC4611, IBM Research, Yorktown Heights, N.Y., November 1973.

4. R.E. Gomory, Large and non-convex problems in linear programming, A.M.S. Symp. Procs. v.15 (1963) 125-139.

5. B.P. Dzielinski and R.E. Gomory, Optimal programming of lot sizes, inventory and labor allocations. Management Science 11 (1965) 874-890.

6. M. Held and R.M. Karp, The traveling-salesman problem and minimum spanning trees. Operations Res. 18 (1970) 1138-1162; ...:part II. Math. Programming 1 (1971) 6-25.

7. M.L. Fisher and J.F. Shapiro, Constructive duality for discrete optimization. Grad. Sch. Bus., U. Chicago, March 1973.

8. G.B. Dantzig and P. Wolfe, The decomposition algorithm for linear programs. Econometrica 29 (1961) 767-778.

9. L. Lasdon, Optimization Theory for Large Systems. Macmillan Co., N.Y. 1970.

10. R.C. Grinold, Steepest ascent for large scale linear programs, SIAM Rev. 14 (1972) 447-464.

11. M. Held, R.M. Karp, and P. Wolfe, Large-scale optimization and relaxation. Proc. 25 Nat. ACM Mtng., Boston 1972.

12. W.W. Hogan, R.E. Marsten, J.W. Blankenship, The BOXSTEP method for large scale optimization. Sloan School of Management, M.I.T., Cambridge, Mass. December 1973.

13. R.T. Rockafellar, Convex Analysis. Princeton U. Press 1970.

14. A. Feuer, Minimization of well-behaved functions, this Conference.

15. V.F. Demjanov, Algorithms for some minimax problems. J. Computer and System Sci. 2 (1968) 342-380.

16. P. Wolfe, Finding the nearest point in a polytope. RC4887, IBM Research, Yorktown Heights, N.Y., June 1974.

17. P. Wolfe, On the convergence of gradient methods under constraint. IBM J. Res. and Devel. 16 (1972) 407-411.

18. D.P. Bertsekas and S. Mitter, A descent numerical method for optimization problems with nondifferentiable cost functionals. SIAM J. Control 11 (1973) 637-652.

19. M. Held, P. Wolfe, and H.P. Crowder, Validation of subgradient optimization. Math. Progr. 6 (1974) 62-88.

20. P. Wolfe, Convergence theory in nonlinear programming. Ch. I of Integer and Nonlinear Programming (J. Abadie, ed.), North-Holland Publ. Co., Amsterdam 1970.

21. H.P. Crowder, Computational improvements for subgradient optimization. RC4907, IBM Research, Yorktown Hts., June 1974.

22. P. Wolfe, A method of conjugate subgradients for minimizing nondifferentiable functions. RC 4857, IBM Res., May 1974.

23. C. Lemarechal, An algorithm for minimizing convex functions. Proc. IFIP Congress 1974, North-Holland Publ. Co., 552-556.

24. C. Lemarechal, An extension of Davidon methods to non differentiable problems. Note, IRIA, Le Chesnay, France 1974.

25. C. Lemarechal, Minimization of nondifferentiable functions with constraints. This Conference.

26. R. Fletcher and M.J.D. Powell, A rapidly convergent descent method for minimization. Computer J. 6 (1963) 163-168.

27. M.R. Hestenes and E. Stiefel, Methods of conjugate gradients for solving linear systems. J. Res. Nat. Bureau of Standards 49 (1952) 409-436.

MINIMIZATION OF NONDIFFERENTIABLE FUNCTIONS WITH CONSTRAINTS

Claude LEMARECHAL

IRIA* - Rocquencourt 78 150 - LE CHESNAY - FRANCE

ABSTRACT

An algorithm is presented, which solves the problem of minimizing a Convex function subject to convex constraints. The nonlinear constraints are exactly penalized, and the linear ones are handled with following the feasible directions idea. The resulting algorithm is closely related to convex analysis ; its main characteristic is that, ε being chosen, the criterion is diminished by ε at each iteration

I - HYPOTHESES - NOTATIONS

Let H be a real Hilbert space. A real function f and a subset B are defined on H, satisfying the fondamental hypotheses :

B is a closed convex set, f is a convex function. (1)

B is contained in the relative interior of the effective domain of f. (2)

The level-sets $\{x \in B / f(x) \leq \alpha\}$ are bounded. (3)

It follows that f is continuous and weakly-lower-semi-continuous on B. For $x \in B$, and $\varepsilon \geq 0$, we define the ε-subdifferential of f at x [7, p. 219] :

$$\partial_\varepsilon f(x) = \{g \in H / \forall y \in H,\ f(y) \geq f(x) + (g, y-x) - \varepsilon\}. \tag{4}$$

which is a closed, bounded, convex set. The indicator function of B is :

$$\delta_B(x) = 0 \text{ if } x \in B, \ +\infty \text{ otherwise} \tag{5}$$

we intend to solve the problem :

$$\min \{f(x)/x \in B\} \tag{6}$$

which is equivalent to

$$\min \{f(x) + \delta_B(x)/x \in H\}. \tag{7}$$

Thus, we are in gross in the framework described in [1], [5], except that the ε-subdifferentials of $f+\delta_B$ are unbounded ; moreover, some further calculations are needed to describe these sets. We recall now the basic results of [5] :

(*) This work forms part of a program of collaborative research between the Electronic Systems Laboratory, M.I.T., Cambridge, Mass., U.S.A. and IRIA-LABORIA, France supported by IRIA-LABORIA and the National Science Foundation, U.S.A. under Grant NSF GK-41647.

Theorem 1.1

Let $x,y \in H$ *and let* $g \in \partial[f+\delta_B](y)$. *Then,* $g \in \partial_\varepsilon[f+\delta_B](x)$ *if and only if* :

$$f(y)+\delta_B(y) \geq f(x) + \delta_B(x) + (g,y-x) - \varepsilon \ . // \qquad (8)$$

Theorem 1.2

Let $\{s^k\}$ *and* $\{g^k\}$ *be two sequences satisfying*

(i) g^k *is bounded.*

(ii) $s^k = \text{Proj}\ \vec{0}/\text{Conv}\{g^1,\dots,g^k\}$.

(iii) $(g^{k+1},s^k) \leq \alpha|s^k|^2$, *with* $\alpha < 1$.

Then, $|s^k| \to 0$. //

As in [1], from a given point $x \in B$ -the current iterate of a feasible direction method- we shall try to construct an ε-descent direction, i.e. a direction s which satisfies :

$$\exists\ \rho > 0 \ \text{ such that } x+\rho s \in B \text{ and } f(x+\rho s) < f(x) - \varepsilon \ .$$

This property is equivalent to : $[f+\delta_B](x+\rho s) < [f+\delta_B](x) - \varepsilon$.

According to (4), it is well-known [1] that :

(i) Either $0 \in \partial_\varepsilon[f+\delta_B](x)$ and, then, x solves (6) within ε, i.e. :

$$\forall\ y \in B,\ f(y) \geq f(x) - \varepsilon \ .$$

(ii) Or $0 \notin \partial_\varepsilon[f+\delta_B](x)$ and, then, the set of all the ε-descent directions is the convex cone defined by :

$$\{s \in H/(s,g) < 0 \quad \forall\ g \in \partial_\varepsilon[f+\delta_B](x)\}.$$

So, the problem is to construct numerically the set $\partial_\varepsilon[f+\delta_B](x)$, and we shall do this in the following way : a finite subset $\{g^1,\dots,g^k\} \subset \partial_\varepsilon[f+\delta_B](x)$ being known, we can compute a "trial ε-descent direction" s^k such that $(s^k,g^i) < 0$ $i=1,\dots,k$. By a study of $f+\delta_B$ along s^k, a further ε-subgradient g^{k+1} is computed, making use of (8) ; the process is iterated until an actual ε-descent direction is obtained. The key-argument is that g^{k+1} can afford some non-negligible information with respect to $\{g^1,\dots,g^k\}$: if s^k and g^{k+1} enter the framework of 1.2, $s^k \in \partial_\varepsilon[f+\delta_B](x)$, and $s^k \to 0$; so, if $0 \notin \partial_\varepsilon[f+\delta_B](x)$, (which is a closed set), the sequence s^k must be finite : for a certain value of k, it is impossible to find $g^{k+1} \in \partial_\varepsilon[f+\delta_B](x)$ such that $(-s^k,g^{k+1}) \leq \alpha|s^k|^2$. If $\alpha \geq 0$, this implies that $(-s^k,g) > 0$ for all $g \in \partial_\varepsilon[f+\delta_B](x)$, so that s^k is really an ε-descent direction.

Thus, the resulting minimization algorithm can be viewed as a numerical implementation of the steepest descent algorithm of Demjanov [3] or, more precisely, of that of Bertsekas and Mitter [1]. It must also be pointed out that the ideas developed here are closely related to those underlying the conjugate subgradient method of [6], [8]. We make an extensive use of the concepts developed in [5].

II - SPECIFICATION OF THE CONSTRAINTS

In (6), let B be defined by

$$B = \{x \in H/(a_j,x) \le b_j \quad j=1,\dots,m$$
$$(a_j,x) = b_j \quad j=m+1,\dots,p$$
$$g_j(x) \le 0 \quad j=p+1,\dots,q\}.$$

Thus, (6) is a problem with m linear inequality constraints, p-m linear equality constraints (g_j are supposed to be convex). Curiously, the easiest constraints to handle with are the nonlinear ones : since the algorithm minimizes nondifferentiable functions, we can penalize the nonlinear constraints by an exact penalty function, following the terminology of [4], in the following way :

$$f_r(x) = f(x) + r \sum_{j=p+1}^{q} \max\,[0,g_j(x)] \qquad r > 0$$

Of course, the hypotheses on g_j must be as those on f, namely : g_j are proper convex continuous functions, the subdifferentials of which are bounded. Under these conditions, Zangwill [9] has shown that there exists $r_o \ge 0$ such that for each $r \ge r_o$, the minimum points of f_r solve (6). Since $f_r^o(x) = f(x)$ if $g_j(x) \le 0$, we have just to minimize f_r for some $r \ge 0$, and increase r if the nonlinear constraints are not satisfied.

Concerning the equality constraints, we can adopt a feasible direction procedure, $x_{n+1}=x_n+\rho_n s_n$, where $(a_j,x_o)=b_j$, and $(a_j,s_n)=0$ for j=m+1,...,p. This technique consists of minimizing f on the linear manifold defined by the linear equality constraints.
So, after an eventual modification of f and H, (6) is in turn equivalent to

$$\begin{cases} \min\ f(x) \\ (a_j,x) \le b_j \qquad j=1,\dots,m. \end{cases} \tag{9}$$

We must point out, however, that these transformations are given for computational convenience only, and that with our algorithm, we can generate feasible points with the nonlinear constraints, as well as we can penalize the linear equality constraints. Henceforth, B is defined as

$$B=\{x\in H/(a_j,x) \le b_j \quad j=1,\dots,m\},$$

and we have the following result [7,p. 283].

Theorem 2.1

If $x \notin B$, $\partial\delta_B(x)=\emptyset$.

If $(a_j,x) < b_j$ $j=1,\dots,m$, then $\partial\delta_B(x) = \{0\}$.

If $(a_j,x) = b_j$ for some j, then $\lambda\, a_j \in \partial\, \delta_B(x) \quad \forall\lambda \ge 0.$ //

So, $\partial\delta_B(x)$ is the convex cone generated by all the binding constraints at x. From [7,th.23.8], we deduce :

Theorem 2.2

Let $x \in B$ be such that $(a_j,x)=b_j$ for some j, and let $g \in \partial f(x)$. Then, for all $\lambda \geq 0$, $g+\lambda\, a_j \in \partial[f+\delta_B](x)$.

III - DETERMINATION OF THE CURRENT DIRECTION s^k.

Our aim here, is to develop the theory for solving the following problem : given $\{g^1,\dots,g^k\} \subset \partial_\varepsilon[f+\delta_B](x)$, find s^k which

- enters the framework of Theorem 1.2.
- is feasible, i.e. $\exists\, \rho_0 > 0$ such that $\rho \in [0,\rho_0] \Longrightarrow x+\rho s^k \in B$.

Let us define the set of all binding constraints at x by :

$$J = \{j=1,\dots,m/(a_j,x) = b_j\}.$$

Then, $\partial\delta_B(x)$ is the convex cone generated by $\{a_j/j\in J\}$. The following result, an extension of 2.2, shows that

$$\{g^1,\dots,g^k\} + \partial\delta_B(x) \subset \partial_\varepsilon[f+\delta_B](x).$$

Theorem 3.1

Let $g \in \partial_\varepsilon[f+\delta_B](x)$. Then, for each set of positive numbers

$$\{\lambda_j/\lambda_j \geq 0,\ j\in J\},\ g+\sum_{j\in J} \lambda_j a_j \in \partial_\varepsilon[f+\delta_B](x).$$

Proof. By hypothesis, we have :

$$\forall\, y\in H,\ f(y) + \delta_B(y) \geq f(x) + \delta_B(x) + (g,y-x) - \varepsilon\ .$$

(i) If $y \in B$, $(a_j,y) \leq b_j = (a_j,x) \quad \forall\, j \in J$.

So, we obtain :

$$(\lambda_j a_j,y-x) \leq 0 \quad \forall\, \lambda_j \geq 0, \quad \forall\, j \in J.$$

Summing all these inequalities, we obtain :

$$\forall\, y\in B,\ f(y) + \delta_B(y) \geq f(x) + \delta_B(x) + (g + \sum_{j\in J} \lambda_j a_j,y-x) - \varepsilon\ .$$

(ii) If $y \notin B$, $\delta_B(y) = +\infty$ and the same inequality holds obviously. //

Thus, paralleling with 1.2. (ii), s^k must be computed as the solution of the problem

$$\min|s|^2\ . \tag{10}$$

$$s = - \sum_{i=1}^{k} \alpha^i g^i - \sum_{j\in J} \lambda_j a_j\ . \tag{11}$$

$$\sum_{i=1}^{k} \alpha^i = 1\ . \tag{12}$$

$$\alpha^i \geq 0 \qquad \lambda_j \geq 0 \tag{13}$$

It can be seen that the unique solution of this problem satisfies

$$\begin{cases} (s^k,-g^i) \geq |s^k|^2 & i=1,\ldots,k \\ (s^k,-a_j) \geq 0 & j \in J \;. \end{cases} \tag{14}$$

But if an iterative method is chosen to solve (10)-(13), this method will generally stop before converging (asymptotic convergence) and (14) may not hold, a fact which could introduce some trouble. Moreover, following the discussion of §2, s^k could have to satisfy some equality constraints to be feasible. So, the form (10)-(13) is unadapted to our purpose. We suggest to solve, instead of it, the following problem :

$$\begin{cases} \min |s|^2 \\ (s,g^i) \leq -1 & i=1,\ldots,k \\ (s,a_j) \leq 0 & j \in J. \end{cases} \tag{15}$$

in which some equality constraints can be added without any difficulty. The following result shows that (10)-(13) and (15) are equivalent :

Theorem 3.2
Either (15) has no feasible solution and $s^k=0$ in (10)-(13). Or (15) has a unique solution $\bar{s}$, and $s^k = \dfrac{\bar{s}}{|\bar{s}|^2}$.

We do not give here the extensive proof, which is out of interest. The key-arguments are :
(i) First, if μ^k is the Lagrange multiplier associated with (12), some classical calculations on the Lagrange function provide

$$\mu^k = |s^k|^2, \text{ and } s^k \text{ solves in fact :}$$

$$\begin{cases} \min |s|^2 \\ (s,g^i) \leq -\mu^k & i=1,\ldots,k \\ (s,a_j) \leq 0 & j \in J \end{cases} \tag{16}$$

(ii) Afterwards, one remarks that (16) involves only homogeneous functions, and, by homothety, its unique solution depends multiplicatively on μ^k ; so, we do not need the exact value of μ^k, which can be replaced by any positive number, say 1, which gives (15).

IV - DETERMINATION OF THE CURRENT ε-SUBGRADIENT g^{k+1}

Suppose we minimize f on B, along s^k. Three possibilities must occur :

(i) One finds $\rho > 0$ such that $x+\rho s^k \in B$ and $f(x+\rho s^k) < f(x)-\varepsilon$.

Then, the current overall iteration of minimization is terminated : s^k was actually an ε-descent direction.

(ii) A second eventuality is that the minimum along s^k is unbinded. We are in this case exactly in the situation described in [5,§4]. In other words, by a dichotomous search, it is possible to determine an ε-subgradient g^{k+1} such that :

$$(s^k,g^{k+1}) \geq -\alpha|s^k|^2, \text{ where } \alpha \in \,]0,1[$$

so that 1.2 applies.

(ii) The third case, in fact the only interesting one here, appears if the unconstrained minimum of $f(x+\rho s^k)$ does not belong to B. It is them impossible to find $g^{k+1} \in \partial_\varepsilon f(x)$ such that $(g^{k+1},s^k) \geq -\alpha|s^k|^2$. Now, we study this case.

We define ρ_{max}, the maximum value for ρ, given by the formula :

$$\rho_{max} = \min_j \{\frac{b_j-(a_j,x)}{(s^k,a_j)} / (s^k,a_j) > 0 \} . \tag{17}$$

In case (iii), $f(x+\rho s^k)$ is decreasing in the neighbourhoud of ρ_{max}, and there exists $j(k)\in\{1,2,\ldots,m\}$ such that

$$(a_{j(k)},x+\rho_{max}s^k) = b_j, \qquad x+\rho_{max}\, s^k \in B.$$

Let $g_{k+1} \in \partial f(x+\rho_{max}s^k)$. From 2.2,

$$g_{k+1} + \lambda a_{j(k)} \in \partial[f+\delta_B](x+\rho_{max}\, s^k), \text{ for all } \lambda \geq 0 .$$

Theorem 4.1

There exists $\lambda_{k+1} > 0$ such that

$$g_{k+1} + \lambda\, a_{j(k)} \in \partial_\varepsilon[f+\delta_B](x) \qquad \forall \lambda \in [0,\lambda_{k+1}]$$

Proof

According to 1.1, $g_{k+1} + \lambda\, a_{j(k)} \in \partial_\varepsilon[f+\delta_B](x)$ if and only if : $\lambda \geq 0$, and $f(x+\rho_{max}s^k) \geq f(x) + \rho_{max}(g_{k+1} + \lambda a_{j(k)},s^k) - \varepsilon$.

So, the desired property is obtained as soon as $\lambda \geq 0$ satisfies :

$$\lambda\rho_{max}(s^k,a_{j(k)}) \leq f(x+\rho_{max}s^k) - f(x)+\varepsilon - \rho_{max}(g_{k+1},s^k) \tag{18}$$

But $f(x+\rho_{max}s^k) - f(x) + \varepsilon \geq 0$ by hypothesis

$(g_{k+1},s^k) < 0$ since the function $f(x+\rho s^k)$ is decreasing,

and $(s^k,a_{j(k)}) > 0$ according to (17).

So, the maximum possible value for λ to satisfy (18) is given by

$$\lambda_{k+1} = \frac{f(x+\rho_{max}s^k)-f(x)+\varepsilon - \rho_{max}(g_{k+1},s^k)}{\rho_{max}(a_{j(k)},s^k)} > 0 . \tag{19}$$ //

Thus, we see that, for continuing the direction-finding procedure, we can add to the set $\{g^1,\ldots,g^k\}$ any point of the form $g_{k+1}+\lambda a_{j(k)}$, $\lambda\in[0,\lambda_{k+1}]$. If λ is too small, 1.2. (iii) will not be satisfied, so we add $g^{k+1} = g_{k+1}+ \lambda_{k+1}\, a_{j(k)}$, and this technique is justified by the following result :

Theorem 4.2

$$(s^k,g^{k+1}) \geq 0 \geq -\alpha|s^k|^2$$

Moreover, the sequence g^{k+1} thus defined is bounded.

Proof.

By definition of λ_{k+1}, we have (see (18)) :

$$(g^{k+1},s^k) = (g_{k+1}+\lambda_{k+1}\ a_{j(k)},s^k) = \frac{f(x+\rho_{max}s^k) - f(x)+\varepsilon}{\rho_{max}} \geq 0 .$$

Now, $|g_{k+1}|$ is bounded (from (2), (3)), $|a_{j(k)}|$ is bounded because it belongs to a finite set. Let us show that λ_{k+1} is bounded. From (19) :

$$|\lambda_{k+1}| \leq \frac{\varepsilon+|g_{k+1}|.|\rho_{max}s^k|}{(a_{j(k)},\rho_{max}s^k)} . \tag{20}$$

The numerator in (20) is bounded (recall (3)). On the other hand,

$$(a_{j(k)},\rho_{max}s^k) = b_{j(k)} - (a_{j(k)},x) \geq 0$$

But since $(a_j,s^k) \leq 0 \quad \forall j \in J$ (see (16)) and from (17), $j(k)$ cannot belong to J : $(a_{j(k)},x) < b_{j(k)}$.

Let $\delta = \inf\{b_j-(a_j,x)/j\notin J\}$. So, $\delta > 0$ is a lower bound for the denominator in (20), and the theorem is proved.

V - AN ALGORITHM

Accordingly, we can propose the following schematic algorithm.

Let $\varepsilon > 0$, $\eta > 0$ be fixed ; let $x_o \in B$. Set n=0.

1. Compute $g^1 \in \partial f(x_n)$; determine J, the set of all binding constraints at x_n. Set k=1.
2. Solve the problem

$$\begin{cases} \min |s|^2 \\ (s,g^i) \leq -1 \quad i=1,\dots,k. \\ (s,a_j) \leq 0 \quad j \in J. \end{cases}$$

If this problem has no feasible solution, or if it has an optimal solution s^k such that $|s^k| \geq \frac{1}{\eta}$ STOP.

3. Determine ρ_{max} according to (17).
4. Minimize $f(x_n+\rho s^k)$ on the interval $[0,\rho_{max}]$, and obtain an optimal ρ^k.

If $f(x_n+\rho^k s^k) < f(x_n) - \varepsilon$, set $x_{n+1} = x_n + \rho^k s^k$, increase n, by 1 and go to step 1.

5. If $\rho^k = \rho_{max}$, determine $g_{k+1} \in \partial f(x_n+\rho^k s^k)$, and a new binding constraint

$j(k) \notin J$. Compute λ_{k+1} by (19), set $g^{k+1} = g_{k+1} + \lambda_{k+1} a_{j(k)}$, increase k by 1 and go to step 2.

6. Otherwise, by a dichotomous search, determine $g^{k+1} \in \partial_\varepsilon f(x_p)$ such that $(g^{k+1}, s^k) \geq - \alpha |s^k|^2$ (see [5,§4] for details), increase k by 1, and go to step 2.//

It is now clear that this algorithm stops after a finite number of iterations at a point $x(\eta)$ such that

$$\forall\, y \in B,\ f(y) \geq f(x(\eta)) - \varepsilon - \eta|y-x(\eta)|.$$

VI - NUMERICAL EXPERIMENTS

A program has been written in Fortran, which solves the problem

$$\begin{cases} \min f(x) & x \in R^n \\ a_i \leq x_i \leq b_i & i=1,\dots,n. \end{cases}$$

In order to be situated with respect to other existing methods, this program has been applied to some problems of Colville [2], the extra constraints being penalized (cf. §2). For each such problem, the algorithm described here appears to be around 50 times slower than the best one. But in front of this poor result, some comments are to be mentioned :

(i) Since almost all the constraints are penalized, it is the technique of [9] which is tested rather than ours.

(ii) The program has been written in a rather crude style ; in particular, the line-search could probably be substantially improved ; but the most important point is the projection of step 2, §5, which spends 90°/° of the computing time ; this problem is solved by the "reduced gradient" method, and it seems that there exists some more efficient methods to solve it.

(iii) Finally, it must be pointed out that our present method has been motivated, not to be fast, but to be fail-safe, and to solve a large class of problems, including "pathological" ones.

In this regard, we can consider that the aim has been reached, giving in particular a contribution to the general decomposition problem where is to be solved

$$\begin{cases} \max h(\lambda) \\ \lambda \geq 0 \end{cases}$$

h, a concave function, being the (generally nondifferentiable) dual function associated to some Lagrange function-see the numerical results in [5].

REFERENCE

[1] D.P. BERTSEKAS, S.K. MITTER, A descent numerical method for optimization problems with non differentiable cost functionals,
SIAM J. on Control 11, n° 4 (1973) pp.637-652.

[2] A.R. COLVILLE, A comparative study on non-linear programming codes,
IBM New York Scientific Center, Techn. Rep. n° 320-2949 (1968).

[3] V.F. DEMJANOV, Algorithms for some minimax problems,
J. of Comp. and Syst. Sc., vol. 2 (1968) pp. 342-380.

[4] J.P. EVANS, F.J. GOULD, J.W. TOLLE, Exact penalty functions in nonlinear programming,
Math. Prog. Vol. 6 n°2 (1969) pp. 299-304.

[5] C. LEMARECHAL, An algorithm for minimizing convex functions,
Proc. IFIP Congress 74 (North Holland) pp. 552-556.

[6] C. LEMARECHAL, Note on an extension of Davidon method to non-differentiable functions,
Math. Prog. (to appear).

[7] R.T. ROCKAFELLAR, Convex Analysis,
Princeton University Press (1970), Princeton, N.J.

[8] P. WOLFE, Note on a method of conjugate subgradients for minimizing non-differentiable functions,
Math. Prog. (to appear).

[9] W.I. ZANGWILL, Non-linear programming via penalty functions,
Man. Sci. Vol. 13, n°5 (1967), pp. 344-358.

MINIMIZING WELL-BEHAVED FUNCTIONS

ALBERT FEUER
MIAMI UNIVERSITY
OXFORD, OHIO

Abstract

Well-behaved objectives such as max functions may be neither convex nor differentiable. Yet few computational methods overcome both these difficulties.

A descent method for a class of "well-behaved" functions which includes both convex and continuously differentiable functions is proposed. Computational experience is described.

INTRODUCTION:

What characterizes a well-behaved function? There are two basic properties: (1) we can easily test a point $\overline{x}\epsilon R^n$ for optimality (2) the function's behavior near $\overline{x}$ determines its behavior at $\overline{x}$. Moreover, there should be a general minimization algorithm applicable to all such functions.

I will describe a generalized derivative which is applicable to well-behaved functions. It will depend upon the derivative at nearby points. A convenient optimality test, based upon a generalization of stationarity is then available. Finally an algorithm which produces such points is given.

FUNDAMENTAL FUNCTIONS:

Def. (1): If $g(x,y): R^n \times R^m \to R$, then $\{g(x,y): y\epsilon S\}$ is a Compact Family of C^1 Functions if (1) S is a compact subset of R^m and (2) $g(x,y)$ and $g_x(x,y)$ are continuous on $R^n \times S$.

A continuously differentiable function is trivially such a family; S then has only a single element.

Def. (2): W( ) is WELL-BEHAVED or a W-FUNCTION, if W( ) is always locally expressible as either the maximum or minimum of a compact family of C^1 functions, i.e. at any point $\overline{x}\epsilon R^n$ there is a positive number $d(\overline{x})$ such that W( ) may be expressed in the closed sphere of radius $d(\overline{x})$ about $\overline{x}$ in one of the following forms.

$U(x) = \max\{g(x,y) : y\epsilon Y(\overline{x})\}$, or $L(x) = \min\{g(x,y) : y\epsilon Y(\overline{x})\}$ where $Y(\overline{x})$ is a compact subspace of Y.

Notes: (1) A continuously differentiable function is Fundamental.

(2) If Y( ) is independent of x, we have the standard objective of a minimax or maximin problem. This leads us to define two special W-Functions.

Def. (3): W( ) is a Standard Max (Min) Function if it may always be expressed locally as the maximum (minimum) of a Compact Family of C^1 Functions.

Note: The absolute value function on the reals may be expressed in the form max{x,-x}; thus it is a Standard Max Function. More generally

Prop. (1): A Finite Convex function is a Standard Max Function.

DIFFERENTIATING WELL-BEHAVED FUNCTIONS:

Def. (4): The CLARKE GENERALIZED GRADIENT of f( ) at a point $\overline{x}\varepsilon X$ is the convex hull of the set of limits of sequences of the form $f(\overline{x} + h^i)$ where $h^i \to 0$. It will be denoted by $CGG_F(\overline{x})$. If it exists, f( ) is called Clarke Differentiable at $\overline{x}$.

F. H. Clarke restricted this definition to Locally Lipschitz Functions, i.e. those which are Lipschitz on all compact sets. Such functions are differentiable a.e. but Clarke Differentiable everywhere [Cl:p.13]. It may be shown that W-Functions are Locally Lipschitz [Fl,Prop.4].

Obs. (1): At a point of continuous differentiability, x, $CGG_W(x)$ coincides with $\nabla W(x)$.

Ex. (1): $W(x) = |x| \quad CGG_W(x) = \begin{cases} -1 & x<0 \\ [-1,1] & x=0 \\ 1 & x>0 \end{cases}$

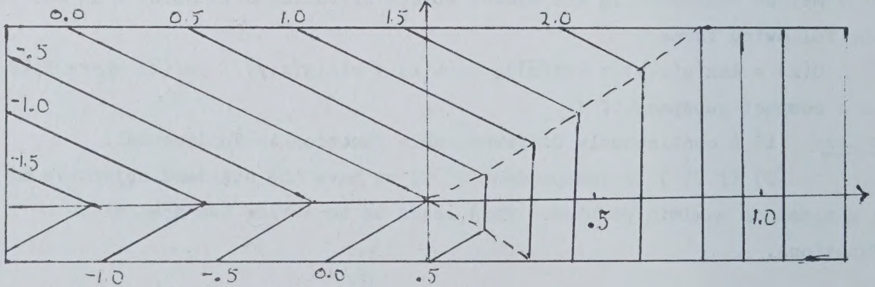
Thus the $CGG_W(\ )$ coincides with the subdifferential of the convex function $|x|$. More generally

Obs. (2): If W( ) is a finite convex function, then $CGG_W(\ )$ equals the subdifferential of W( ) [Rl,Thm.25.6].

Def. (5): For a W-Function, $A(\overline{x})$, the set of Active Components at X is $\{y\varepsilon Y(\overline{x}): g(\overline{x},y) = W(\overline{x})\}$. A single active component determines W(x). Theorem (1) will show that the set of active components determines $CGG_W(x)$.

Theorem (1): For a W-Function, $CGG_W(x^*) = CH\{g_x(x^*,y):y\varepsilon A(x^*)\}$ where CH(S) denotes the convex hull of the set S.

Ex. (2): $W(x) = \text{Max}\{x_1 + 2x_2, x_1 - 2x_2, 3x_1 + x_1^3\}$



The first function is active in the top region, the second in the bottom region and the third in the remainder. At (1,1.5) the first and third functions are active; therefore, $CGG_W(1,1.5) = CH\{(1,2),(6,0)\}$. At (0,0) all three functions are active; thus $CGG_W(0,0) = CH\{(1,2),(1,-2),(3,0)\}$ which is a triangle not enclosing the origin.

Theorem (2): For a W-Function expressible near x* as a Max [Min] Function, $W(x^*;d) = \text{Max [Min]}\{g_x(x^*,y)\cdot d : y \varepsilon A(x^*)\}$

Note: At a point of continuous differentiability, both expressions reduce to $\nabla W(x^*)\cdot d$.

THE MINIMIZATION OF WELL-BEHAVED FUNCTIONS:

Interior Extremum Theorem: If x is a local extremum of a W-Function, W( ), then $0 \varepsilon CGG_F(x)$.

Note: This was first shown by V. F. Demyanov in 1966 for the Standard Minimax objectives, [D1,p.50,Note 1].

Def. (6): $\overline{x}$ is a Stationary Point of a Clarke Differentiable Objective W( ), then $0 \varepsilon CGG_F(\overline{x})$.

Obs. (3): If W( ) is both stationary and continuously differentiable at x*, then $\nabla W(x^*) = 0$.

This paper's algorithm stems fromthe most classical minimization technique - Cauchy's Steepest Descent Method.

Def. (7): A Steepest Descent Direction (SDD) at x of f() [B p.347] is a vector y(x) such that $f'(x;y(x)) = \min\{f'(x;y) : |y| \leq 1\}$.

At a point of differentiability, the SDD is the negative of the gradient. More generally

Prop. (2): If W( ) is a W-Function locally expressible as a max function and x* is not stationary, then (a) the SDD of W( ) at x* is $-m/|m|$ where m is the vector of minimal norm in $CGG_W(x^*)$, (b) $F'(x^*;d) \geq 0$, if $d \varepsilon CGG_W(x^*)$.

Prop. (2*): If W( ) is a W-Function locally expressible as a min Function and x* is not stationary, then (a) a SDD of W( ) at x* is $M/|M|$ where M is a vector of maximal norm in $CGG_W(x)$ (b) $W'(x^*;-d) \leq 0$, if $d \varepsilon CGG_W(x^*)$

It may be shown that Cauchy's Method Fails for Ex. (2) [C,p.21]. All proofs of the validity of Cauchy's Method are based upon the implication:

$$x_i \to x^* \Longrightarrow F'(x_i) \to F'(x^*).$$

The corresponding statement for the CGG of an W-Function is untrue. E.g. look at Ex. (2). As $x \to (0,0)$ along the upper boundary curve $CGG_W(x) \to CH\{(1,2),(3,0)\}$ which is a proper subset of $CGG_F(0,0)$.

A convergent algorithm for well-behaved objectives must anticipate the CGG at a limit point. Cauchy's Method which ignores data from points other than the current one does not do this.

This paper's method, by contrast forms an approximate CGG at each point of the sequence using data from the most recent points of the sequence. After checking its accuracy we proceed to a new point in the pseudo-steepest descent direction - the negative of the vector of minimal norm in the current approximation. All accumulation points are stationary.

THE STEP-SIZE AIM AND THE AUXILIARY FUNCTIONS

A. A. Goldstein first introduced "serious" steps in the context of Cauchy's Method [Gl]. P. Wolfe [Wl] introduced the term and broadened the concept to include other methods.

Def. (8): A step-size t in the direction d at x* is SERIOUS for a directionally differentiable Function f( ) if

$$(1)\ f'(x^*;d)<0$$

$$(2)\ \frac{f(x^* + td) - f(x^*)}{t} \geq E\cdot f'(x^*;d)$$

$$(3)\ f'(x^* + td;d) \leq (1-E)\cdot f'(x^*;d)$$

where E is a positive constant less than 1/2.

These conditions suggest a minimization strategy. By Condition (1), f( ) decreased in the direction 'td' for small 't'; thus proceed along the ray d AS LONG AS f( ) decreases significantly at which point we repeat the process with a new descent direction.

We make this precise by calling a decrease "significant" if it is at least as great per unit of "d" as some fixed positive fraction of the initial directional derivative (Condition (2)).

If there is no upper bound on t, the function must be unbounded below. Otherwise there would be a point on the ray x* + td at which the rate of decrease is significantly less than its original value (Condition (3)).

Theorem (2): For a W-Function, W( ), locally expressible as a max [min] function $W'(x^*;d) = \max\ [\min]\{t\cdot d : t \in CGG_W(x^*)\}$.

Obs. (4): To implement the algorithm, we must be able to calculate (x) and an element of $CGG_W(x)$ for arbitrary x and to recognize the type of local representation applicable on some small interval.

Several minimization methods for nondifferentiable functions [Bl, D2,D3] require all of $CGG_W(x)$; but Y(x)'s size often precludes its generation. Under my more limited assumptions, we can't in general get F'(x;d) [Thm.2]. We overcome this handicap with two auxiliary functions.

Def. (9): If W( ) is Well-Behaved, then g(t) denotes the element of $CGG_W(x^* + td)$ which we produce and

$$M(t) \equiv \frac{\langle g(t),d\rangle}{|d|^2}, \quad Q(t) \equiv \frac{W(x^*+td) - W(x^*)}{t|d|^2} \quad t \geq 0$$

Note: $M(\ )$ resembles a directional derivative and $Q(\ )$ resembles a difference quotient.

Obs. (5): If $W(\ )$ is a W-Function differentiable along $\{x^*+td : t \geq 0\}$ and $d = -W'(x^*) \neq 0$, then the serious conditions correspond to

(S1) $W'(x^*;d) = -|d|^2 < 0$ (S2) $Q(t) \leq -E$ (S3) $M(t) \geq E-1$

Def. (10): A step-size t in the direction d at $\overline{x}$ is SERIOUS for a W-Function if (1) $Q(t) \leq -S1$ and (2) $M(t) \geq -S2$, where $0 < S1 < S2 < 1/2$

THE ALGORITHM:

Let us bring the previous ideas together by describing the algorithm in some detail. Recall the aim was to produce a stationary point, i.e., a point at which the vector of minimal norm in its CGG is zero.

After finitely many iterations, x will become "almost stationary" i.e. we have an accurate approximate CGG at x which contains a vector of small norm. We then tighten the relevant accuracy and smallness criteria and restart from x.

Notation: B: an a priori lower bound on the objective

APP_k: the set of vectors whose convex hull approximates $CGG_F(x)$

W_k: the negative of the vector of minimal norm in the current approximation.

A: the "dispersion" of APP_k's spanning points, those points at which elements of APP_k are generated.

The user need choose only B and initial X; $(\Delta,\varepsilon,S1,S2)$ is preset with $0 < S1 < S2$

A TYPICAL ITERATION

PHASE (I): Test for Almost Stationarity

(1) If $|d_k| \geq \varepsilon$, go to (3). This occurs when either the current approximation requires augmentation or we can generate a serious step in the direction d_k. We make this determination in Phase (II).

(2) Thus $|d_k| < \varepsilon$. If $A \leq \Delta$, we have almost stationarity. Reduce the tolerances (ε,Δ) and restart from X_k. Otherwise the norm of d_k is small but the spanning points are scattered. The result is ambiguous, we have gleaned all the useful data from the current approximation. Discard it and begin anew from X_k.

PHASE (II): Check the Approximation's Accuracy

(3) Find $T^* > 0$ satisfying one of the following

(a) $W(X_k + T^*d_k)<B$ STOP. Something is amiss, the function is below the given lower bound.

(b) $|T^*d_k| \leq \Delta$, $Q(T^*)>-S1$, $M(T^*) \geq -S2$

We have a point Δ-close to X_k at which the function has not decreased significanlty, although its directional derivative has increased significantly. Thus the representative of its CGG differs significantly from the convex hull of APP_k. Therefore the approximation should be augmented using data from the nearby point $(X_k + T^*d_k)$

(c) $Q(T^*) \leq -S1$, $M(T^*) \geq -S2$

This is a serious step-size, with both a significant decrease in W(x) and a significant increase in the directional derivative.

In short, if the lower bound is correct we verify the accuracy of APP_k by trying to generate a serious step-size. If we are successful the accuracy is satisfactory, otherwise we produce a nearby point which is used to improve the approximation.

PHASE (III): Update X and the Approximate CGG

(4) $APP_{k+1} = APP_k \cup CGG_W(X_k + T^*d_k)$
$A = \text{Max}\{A, |X_k + T^*d_k - XO|\}$.

(5) If we have come from (3b), let $X_{k+1} = X_k$ and return to (1). Otherwise $X_{k+1} = X_k + T^*d_k$. Return to (1), if $F(X_{k+1})<B$. If not, STOP.

In short, we update the approximate CGG and the measure of dispersion (the maximal distance from a spanning point X_k to the oldest spanning point XO). The step-size type determines where the new approximation applies. If it was serious it applies to the new X; otherwise at the old X.

SAMPLE CALCULATIONS:

A simple example will clarify many of these operations.

Ex. (3): $W(X) = |X_1| + |X_2| = \text{Max}\{X_1+X_2, -X_1+X_2, -X_1-X_2, X_1-X_2\}$

Let $S1 = .4, S2=.45$, $\Delta=\varepsilon=.01$, and $B=-1$. For the sake of clarity the actual calculations for the step-size will be omitted. Start at (2,2).

$X_1=XO=(2,2)$ $A=0$ $APP_1=(1,1) \Rightarrow d_1=(-1,-1)$

2.5 is a Serious Step

$X_2=(-1/2,-1/2)$ $A=5/\sqrt{2}$ $APP_2=\{(1,1),(-1,-1)\} \Rightarrow d_2=(0,0)$

Ambiguous Result $\therefore$ Reset

$XO=(-1/2,-1/2)$ $A=0$ $APP_2=(-1,-1) \Rightarrow d_2=(1,1)$

Look at $T=1/2$ $X=(0,0)$. For the first time the choice of g(STEP) is unclear. $CGG\{(0,0)=CH\{(1,1),(1,-1),(-1,1),(-1,-1)\}$. Choose $g(1/2)=(1,1)$ $M(1/2)=0$. We have a serious step-size.

$X_3 = (0,0) \quad A = 1/\sqrt{2} \quad APP_3 = \{(-1,-1),(1,-1)\} \Rightarrow d_3 = (0.1)$

A Serious Step significantly reduces the objective. (0,0) is the global minimizing solution; so any decrease, let alone a significant one, is impossible. Thus the next point must be minor, it is on the positive X_1-Axis and within .01 of (0,0). For any choice of g(STEP), M(STEP)=1. Thus X remains fixed.

$X_4 = (0,0) \quad A = 1/\sqrt{2} \quad APP_4 = \{(-1,-1),(1,-1),(1,1)\} \Rightarrow d_4 = (0,0)$

Ambiguous Result $\therefore$ Reset

$X0 = (0,0) \quad A = 0 \quad APP_4 = \{(1,-1)\} \Rightarrow d_4 = (-1,1)$

A Serious Step is again impossible. The next point is minor and in the fourth quadrant; X will remain fixed.

$X_5 = (0,0) \quad A \leq .01 \quad APP_5 = \{(1,-1),(-1,1)\} \Rightarrow d_5 = (0,0)$

We now have Almost Stationarity: a small vector is in the approximation and the spanning vectors are tightly clustered. Moreover, the algorithm's termination criteria is satisfied, so we are finished.

THE ALGORITHM'S IMPLEMENTATION

It may be shown that each task of the algorithm requires only finitely many function evaluations. I tested the two versious of the algorithm, FIRE and EFIRE, on ten of Dixon's test functions [D4] and achieved convergence with seven to twelve times the number of functionsl evaluations used by the best conventional techniques. Knowing of no standard test functions, I chose my own. On these, the results were also satisfactory.

TABLE (1)

FUNCTION EVALUATIONS UNTIL TERMINATION

	B	W	F	F_R	E	E_R
1)	108	210	716	6.62	1239	11.47
2)	41	80	195	4.75	224	5.46
3)	47	67	163	3.46	209	4.45
4)	64	128	393	6.14	575	8.98
5)	147	305	1300	8.84	1351	9.19
6)	349	1058	2056	5.89	3935	11.27
7)	160	333	1213	7.58	4488	28.05
8)	235	705	3175	13.51	6001	25.53
9)	60	150	292	4.86	277	4.62
10)	536	1890	3929	7.32	3553	6.62

B: Best Response Reported by Dixon

W: Worst Response Reported by Dixon

F: FIRE's Results

F_R: FIRE's Result/B's Result

E: EFIRE's Results

REFERENCES

B2. Bertsekas, Dimitri P. and Mitter, Sanjoy K., "Steepest Descent for Optimization Problems with Nondifferentiable Cost Functionals."

Proceedings of the Fifth Annual Princeton Conference on Information Sciences, March 25, 1971.

C1. Clarke, Frank H., Necessary Condition for Nonsmooth Problems in Optimal Control and the Calculus of Variations, Ph.D. Thesis, Department of Mathematics, University of Washington, 1973.

C2. Cullum, Jane, Donath, W. E., Wolfe, Philip, "An Algorithm for Minimizing Certain Nondifferentiable Convex Functions," IBM Research Report RC-4611. November 6, 1974.

D1. Demyanov, Vladimir F., "On the Solution of Several Minimax Problems, I." Kibernetika, Vol. 2, No. 6, pp. 58-66, 1966.

D2. —,"Algorithms for Some Minimax Problems," Journal of Computer and System Sciences, Vol. 2, pp. 342-380, 1968.

D3. — and Malozemov, V. N. "On the Theory of Nonlinear Minimax Problems," Russian Math. Surveys, Vol. 26, pp. 57-115, 1971.

D4. Dixon, L.C.W., "Conjugate Directions without Linear Searches," J. Inst. Maths. Applics., Vol. 11, pp. 317-328, 1973.

F1. Feuer, Albert, An Implementable Mathematical Programming Algorithm for Admissible Fundamental Functions, Ph.D. Thesis, Department of Mathematics, Columbia University, 1974.

G1. Goldstein, A. A., Minimizing functionals on normed-linear spaces," J. SIAM Control, Vol. 4, pp. 81-89, 1966.

R1. Rockafellar, R. Tyrrel, Convex Analysis, Princeton, Princeton University Press, 1970.

W1. Wolfe, Philip, "Convergence Conditions for Ascent Methods," SIAM Review, Vol. 11, No. 2, pp. 226-235, 1969.

W2.—, "A Method of Conjugate Subgradients for Minimizing Nondifferentiable Functions, IBM Research Report RC-4857, May 22, 1974.

APPENDIX

TEST FUNCTIONS:

1) ROS2: $F = 100(X_1^2 - X_2)^2 + (1 - X_1)^2$
$XO = (-1.2,1)$

2) BOX2: $F = \Sigma_{K=1}^{10}(e^{-x_1 z_K} - e^{-x_2 z_K} - e^{-z_K} + e^{-10 z_K})^2$ where $z_K = K/10$
$XO = (5,0)$

3) EXP2: $F = \Sigma_{K=1}^{10}(e^{-x_1 z_K} - 5e^{-x_2 z_K} - e^{-z_K} + 5e^{-10 z_K})^2$ where $z_K = K/10$
$XO = (1,2)$

4) EXP3: $F = \Sigma_{K=1}^{10}(e^{-x_1 z_K} - x_3 e^{-x_2 z_K} - e^{-z_K} + 5e^{-10 z_K})^2$ where $z_K = K/10$
$XO = (1,2,1)$

5) EXP4: $F = \Sigma_{K=1}^{10}(x_3 e^{-x_1 z_K} - x_4\ 3e^{-x_2 z_K} - e^{-z_K} + 5e^{-10 z_K})^2$ where $z_K = K/10$
$XO = (1,2,1,1)$

6) EXP5: $F = \Sigma_{K=1}^{10}(x_3 e^{-x_1 z_K} - x_4\ e^{-x_2 z_K} + 3e^{-x_5 z_K} - e^{-z_K} + 5e^{-10 z_K} -$
$3e^{-4 z_K})^2$ where $Z_K = K/10$
$XO = (1,1,1,1,1)$

7) POW4: $F = (X_1 + 10X_2)^2 + 5(X_3 - X_4)^2 + (X_2 - 2X_3)^4 + 10(X_1 - X_4)^4$
$XO = (3,-1,0,1)$

8) WOOD4: $F = 100(X_2 - X_1^2)^2 + (1 - X_1)^2 + 90(X_4 - X_3^2)^2 + (1 - X_3)^2 +$
$10.1((X_2 - 1)^2 + (X_4 - 1)^2) + 19.8(X_2 - 1)(X_4 - 1)$
$XO = (-3,-1,-3,-1)$

9) ROS8: $F = 100(X_2 - X_1^2)^8 + (1 - X_1)^8$
$XO = (-1.2,1)$

10) TEST10: $F = (1 - X_1)^2 + (1 - X_{10})^2 + \Sigma_{K=1}^{9}(X_K^2 - X_{K+1})^2$
$XO = (-2,-2,-2,-2,-2,-2,-2,-2,-2,-2)$

AN ALGORITHM FOR THE SOLUTION OF A NONDIFFERENTIABLE MINIMIZATION PROBLEM THAT AROSE IN GRAPH PARTITIONING STUDIES

JANE CULLUM, W. E. DONATH, P. WOLFE
IBM Thomas J. Watson Research Center, Yorktown Heights, N.Y.

SUMMARY

For any given graph $G = \{V,E\}$, of interest are partitions of the set of vertices V into q subsets of equal size for which the number of edges in E connecting different subsets of the partition is minimal. Realization of such a minimal partition is generally not possible and a heuristic, suboptimal partition must be used.

Donath and Hoffman [1] have derived a lower bound on the number of edges connecting the subsets resulting from any partition of V into q subsets of equal size. This bound is obtained from the minimum value of the sum of the q algebraically largest eigenvalues of a modified adjacency matrix of the given graph, where only the diagonal entries of this matrix are allowed to vary and the trace of the matrix is held constant. They have also shown that the eigenelement configuration at a minimizing point of the above minimization problem can be used to obtain a heuristic partition [2].

In this talk, relevant properties of the sum of the q algebraically largest eigenvalues of any real symmetric matrix as a function of the diagonal entries are first discussed. In particular, any such sum is a convex function of the diagonal entries, but it is not necessarily everywhere differentiable. Points of nondifferentiability are identified and a computable characterization of the subdifferential of any such sum at any given point is derived. A convergent, implementable procedure for solving the above minimization problem is then presented. Detailed comments and proofs and numerical results are in reference [3].

Existing procedures Schor [4], Demyanov [5], and Bertsekas and Mitter [6] for the minimization of a general convex but nondifferentiable function were not suitable for this problem. The procedure in [4] requires the user to specify a strategy for selecting, at each iteration, a direction of search from the subdifferential and an appropriate step along the direction chosen. The procedures in [5] and [6] on the other hand require at each iteration computationally feasible characterizations of specific enlargements of the subdifferential. For the given problem neither a strategy nor the required characterizations of the associated enlarged subdifferentials could be easily determined. The algorithm proposed is similar to those in [5] and [6]; however, the enlarged subdifferentials that it utilizes are natural enlargements of the subdifferentials directly related to the cause of nondifferentiability. The computable characterizations obtained for the original subdifferentials can be easily extended to these particular enlargements.

The development uses results from eigenelement analysis and from optimization theory.

REFERENCES

1. W. E. Donath and A. J. Hoffman, Lower bounds for the partitioning of graphs, IBM J. Res. and Dev. 17 (5), (September 1973), pp. 420-425.
2. W. E. Donath and A. J. Hoffman, Algorithms for partitioning of graphs and computer logic based on eigenvectors of connection matrices, IBM Technical Disclosure Bulletin 15 (3), (August 1972), pp. 938-944.
3. Jane Cullum, W. E. Donath and P. Wolfe, An algorithm for minimizing certain nondifferentiable convex functions, IBM RC 4611, November 1973.

4. Ju. Ermol'ev, Methods of solution of nonlinear extremal problems, Kiberneticka 4, (1966), pp. 1-14.
5. V. F. Demyanov, Seeking a minimax on a bounded set. Soviet Math. Doklady 11, pp. 517-521; trans of Dokl. Akad. Nauk SSSR 191, (1970).
6. D. P. Bertsekas and S. K. Mitter, Steepest descent for optimization problems with nondifferentiable cost functions. Proc. 5th Annual Princeton Conf. on Information Sciences and Systems, 1971.

ON GRADIENT PROJECTION

A. A. GOLDSTEIN
Professor of Mathematics, Department of Mathematics
University of Washington, Seattle, Washington 98195

In a recent work [1], D. P. Bertsekas proposed a variation of the gradient projection method of [2]. We propose a similar variation on a restricted class of convex sets, and also discuss a Newton type procedure.

Let C be a closed convex set in Hilbert space, H. Let P denote the mapping (Projection) which assigns to each point in H its closest point in C. The right-handed variation of a mapping g at the point x in the direction h is denoted by $g_+'(x,h)$ whenever $\lim\limits_{t \to 0+} \frac{g(x+th) - g(x)}{t}$ exists. A point $x \in C$ is called stationary for f iff $P(x-\nabla f(x)) = x$. Equivalently, x minimizes $[\nabla f(x),\cdot]$ on C. Suppose $P_+'(x,-\nabla f)$ exists for $x \in C$. It is immediate that $P_+'(x,-\nabla f) = 0$ is a necessary condition for x to be a stationary point. For certain sets C, e.g. polyhedral sets and sets with unique continuously turning support hyperplanes, the condition is also sufficient. We call a closed convex set simple if for every f with ∇f defined on C, $P_+'(x,-\nabla f)$ exists and $P_+'(x,-\nabla f) = 0$ iff x is stationary.

Theorem. Assume f is defined on C. For arbitrary $x_0 \in C$, let $S = \{x \in C: f(x) \leq f(x_0)\}$. Assume ∇f is defined on S.

1) If C is simple, if f is bounded below, and if ∇f is uniformly continuous on C, then sequences $\{x_k\} \subset C$ and $\{\gamma_k\} \subset R$ can be constructed with $\{\gamma_k\}$ bounded away from 0 such that $\{x_k - P(x_k - \gamma_k \nabla f(x_k)\} \to 0$ and $f(x_{k+1}) < f(x_k)$. If in addition S is compact and z a cluster point of $\{x_k\}$, then z is stationary. If the number of stationary points are finite, $\{x_k\}$ converges.

2) Assume f'' exists on S; $|f''(x)| \leq M$ for all $x \in S$; $\nabla f(x) \neq 0$, $x \in C$; $|f''(x) - f''(y)| \leq L_1 |x - y|$ for x,y in S; P is Frechet differentiable on $S_M = \{x - \frac{1}{M} f(x): x \in \delta C\}$ and $|P'(x) - P'(y)| \leq L_2 |x-y|$ for x,y S_M. If z is stationary assume that $f''(z,k,h) \geq N|h|^2$ for all $h \in H$ and some $N > 0$. Assume either that S is compact and the number of stationary points is finite, or $f''(x,h,k) \geq N|h|^2$ for all $x \in S$ and $h \in H$.

Then a sequence $\{x_k\}$ can be constructed which converges to z. The ultimate rate of convergence is quadratic. The point z is a local minimum, or global minimum if $f(x,h,k) \geq N|h|^2$ for all $x \epsilon S$.

Proof of 1) Set $x(\gamma) = P(x - \gamma \nabla f(x))$ and $g(x,\gamma) = \frac{f(x)-f(x(\gamma))}{[\nabla f(x), x-x(\gamma)]}$. Choose δ, $0 < \delta \leq 1/2$. The sequence $\{x_k\}$ is constructed inductively as follows. Take x_0 arbitrarily in C. Assume x_k is given. If $g(x_k,1) < \delta$ choose γ_k so that $\delta \leq g(x_k,\gamma_k) \leq 1 - \delta$; otherwise set $\gamma_k = 1$. Set $x_{k+1} = P(x_k - \gamma_k \nabla f(x_k))$. We now show that γ_k can be chosen as claimed.

By the mean value theorem:

$$f(x) - f(x(\gamma)) = [\nabla f(x), x - x(\gamma)] + [\nabla f(\xi) - \nabla f(x), x - x(\gamma)]$$

with ξ between x and $x(\gamma)$. Thus

$$g(x,\gamma) = 1 + \frac{[\nabla f(\xi) - \nabla f(x), \frac{x(\gamma) - x}{\|x(\gamma) - x\|}]}{[\nabla f(x), \frac{x(\gamma) - x}{\|x(\gamma) - x\|}]} \qquad \text{(I)}$$

By the separation theorem $[\gamma \nabla f(x), x(\gamma) - x] \geq \|x(\gamma) - x\|^2$

whence $[\nabla f(x), \frac{x(\gamma) - x}{\|x(\gamma) - x\|}] \geq \|\frac{P(x - \gamma \nabla f(x)) - P(x)}{\gamma}\|$

If x is not stationary,

$$\liminf_{\gamma \to 0_+} [\nabla f(x), \frac{x(\gamma) - x}{\|x(\gamma) - x\|}] \geq \|P'_+(x, -\nabla f(x))\| > 0$$

Thus $\lim_{\gamma \to 0_+} g(x,\gamma) = 1$. If $g(x,1) < \delta$, then by continuity g achieves all values between δ and $1-\delta$. Clearly $x_{k+1} \epsilon C$. Since $f(x_k) - f(x_{k+1}) = g(x_k,\gamma_k)[\nabla f(x_k), x_k - x_{k+1}] \geq \frac{\delta}{\gamma_k} \|x_{k+1} - x_k\|^2$, $f(x_k) < f(x_{k+1})$.

Because $\|\xi_k - x_k\| \leq \|x_{k+1} - x_k\| = \|P(x_k - \gamma_k \nabla f(x_k)) - x_k\|$, if $\{\gamma_k\} \to 0$ so does $\{\xi_k - x_k\}$. By the uniform continuity of ∇f we see that $\{g(x_k,\gamma_k)\} \to 1$, a contradiction. Hence γ_k is bounded away from 0, and bounded above by 1. If $\{x_{k+1} - x_k\} \not\to 0$, we contradict that f is bounded below on C. If the cluster points of $\{x_k\}$ are finite in number then since $\{x_{k+1} - x_k\} \to 0$, it follows that $\{x_k\}$ converges.

Proof of 2) We first construct a sequence converging to z without assuming C is simple. Since $|\nabla f(x) - \nabla f(y)| \leq M\|x - y\|$ $x,y \epsilon C$, we may rewrite (I) as follows

$$g(x,\gamma) \geq 1 - \gamma M \qquad (I').$$

Then the construction of $\{x_k\}$ is as above, mutatis mutandis. Now $f''(x,h,h) > 0$ in some neighborhood of z because of the Lipschitz condition on f''. Hence for some k, $f''(x_k,h,h) \geq N_k|h|^2$, for all $h\epsilon H$, for some $N_k > 0$. Moreover $\{N_k\} \rightarrow N$. Set $\eta_k = x_k - P(x_k - \frac{1}{M}\nabla f(x_k))$. If z is stationary $z - P(z - \frac{1}{M}\nabla f(z)) = 0$. Hence $\{\eta_k\} \rightarrow 0$. Set $h_k = \frac{1}{N_k^2}(\frac{1}{M}L_1 + L_2(3 + \frac{1}{M}\eta_k L_1)^2\eta_k M^2$. Then $\{h_k\} \rightarrow 0$. We choose k so large that $h_k \leq \frac{1}{2}$ and $N_k > 0$. Now define $T(x) = x - P(x - \frac{1}{M}\nabla f(x))$ and $x_{k+1} = x_k - (T'(x_k))^{-1}T(x_k)$. By [3], $(T'(x_j))^{-1}$ exists, $j = k, k+1, \ldots$ and $\{x_k\} \rightarrow z$ with quadratic convergence, because the conditions of Kantorovich's theorem are fulfilled.

BIBLIOGRAPHY

[1] Dimitri P. Bertsekas, On the Goldstein-Levitan-Polyak Gradient Projection Method. To appear.

[2] A. A. Goldstein, Convex Programming in Hilbert Space, Bull. A.M.S., Vol. 70 #5, pp. 709-710, 1964.

[3] A. A. Goldstein and R. A. Tapia, Kantorovich Estimates for Some Convex Programs, SIAM J. Control, Vol. 11 #2, 1973, pp. 375-377.

[4] J. B. Kruskal, Two Convex Counterexamples. A discontinuous envelope function and a nondifferentiable nearest point mapping, PAMS Vol. 23 #3, Dec. 1969, pp. 697-703.

NONDIFFERENTIABLE OPTIMIZATION VIA APPROXIMATION[*]

DIMITRI P. BERTSEKAS
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois 61801

ABSTRACT

This paper presents a systematic approach for minimization of a wide class of nondifferentiable functions. The technique is based on approximation of the nondifferentiable function by a smooth function and is related to penalty and multiplier methods for constrained minimization. Some convergence results are given and the method is illustrated by means of examples from nonlinear programming.

1. INTRODUCTION

Optimization problems with nondifferentiable cost functionals, particularly minimax problems, have received considerable attention recently since they arise naturally in a variety of contexts. Optimality conditions for such problems have been derived by several authors while a number of computational methods have been proposed for their solution (the reader is referred to [1] for a fairly complete list of references up to 1973). Among the computational algorithms currently available are the subgradient methods of [19,15,10], the ϵ-subgradient method [1,2] coupled with an interesting implementation of the direction finding step given in [12], the minimax methods of [6,7,9,17] which were among the first proposed in the nondifferentiable area, and the recent interesting methods proposed in [5,20]. While the advances in the area of computational algorithms have been significant, the methods mentioned above are by no means capable of handling all problems encountered in practice since they are often limited in their scope by assumptions such as convexity or they are applicable only to a special class of problems such as minimax problems of particular form. Furthermore several of these methods are similar in their behavior to either the method of steepest descent or first order methods of feasible directions and converge slowly when faced with relatively ill-conditioned problems. Thus there is considerable room for new methods and approaches for solution of nondifferentiable problems and the purpose of this paper is to provide a class of methods which is simple to implement, is quite general in its scope, and relies on an entirely different philosophy than those underlying methods already available.

We consider minimization problems of the form

$$\text{minimize } g(x) \quad \text{subject to } x \in Q \subset R^n \tag{1}$$

where g is a real valued function on R^n (n-dimensional Euclidean space).

---

[*] This work was supported in part by the Joint Services Electronics Program (U. S. Army, U. S. Navy, and U. S. Air Force) under Contract DAAB-07-72-C-0259, and in part by the U. S. Air Force under Grant AFOSR-73-2570.

We consider the case where the objective function g is nondifferentiable exclusively due to the presence of several terms of the form

$$\gamma[f_i(x)] = \max\{0, f_i(x)\} \qquad i \in I \tag{2}$$

where $\{f_i | i \in I\}$ is an arbitrary collection of real valued functions on R^n. By this we mean that if the terms $\gamma(\cdot)$ in the functional expression of g were replaced by some continuously differentiable functions $\tilde{\gamma}(\cdot)$ then the resulting function would also be everywhere continuously differentiable.

For purposes of easy reference we shall call a term of the form (2), a simple kink. It should be emphasized that while we concentrate attention on simple kinks, the approach is quite general since we do not necessarily require that the functions f_i in (2) are differentiable but rather we allow them to contain in their functional expressions other simple kinks. In this way some other kinds of nondifferentiable terms such as for example terms of the form

$$\max\{f_1(x), \ldots, f_m(x)\} \tag{3}$$

can be expressed in terms of simple kinks by writing

$$\max\{f_1, \ldots, f_m\} = f_1 + \gamma[f_2 - f_1 + \gamma[\ldots\gamma[f_{m-1} - f_{m-2} + \gamma[f_m - f_{m-1}]]]]. \tag{4}$$

Since there are no restrictions on the manner in which simple kinks enter in the functional expression of g, a little reflection should convince the reader that the class of nondifferentiable problems that we are considering is indeed quite general.

The basic idea of our approach for numerical solution of problems of the form (1) is to approximate every simple kink in the functional expression of g by a smooth function and solve the resulting differentiable problem by conventional methods. In this way an approximate solution of problem (1) will be obtained which hopefully converges to an exact solution as the approximation of the simple kinks becomes more and more accurate.

While, as will be explained shortly, other approximation methods are possible, we shall concentrate on the following two-parameter approximation $\tilde{\gamma}[f(x),y,c]$ of a simple kink $\gamma[f(x)]$

$$\tilde{\gamma}[f(x),y,c] = \begin{cases} f(x) - \dfrac{(1-y)^2}{2c} & \text{if } \dfrac{1-y}{c} \le f(x) \\ y f(x) + \dfrac{c}{2}[f(x)]^2 & \text{if } -\dfrac{y}{c} \le f(x) \le \dfrac{1-y}{c} \\ -\dfrac{y^2}{2c} & \text{if } f(x) \le -\dfrac{y}{c} \end{cases} \tag{5}$$

where y,c are parameters with

$$0 \le y \le 1 \quad , \; 0 < c. \tag{6}$$

If the function f is differentiable then the function $\tilde{\gamma}[f(x),y,c]$ above is also differentiable with respect to x. Its gradient is given by

$$\nabla\tilde{\gamma}[f(x),y,c] = \begin{cases} \nabla f(x) & \text{if} \quad \frac{1-y}{c} \leq f(x) \\ [y + cf(x)]\nabla f(x) & \text{if} \quad -\frac{y}{c} \leq f(x) \leq \frac{1-y}{c} \\ 0 & \text{if} \quad f(x) \leq -\frac{y}{c} \end{cases} \tag{7}$$

The functional form of $\tilde{\gamma}$ is depicted in Figure 1. It may be seen that

$$\begin{aligned} \tilde{\gamma}(t,y,c) \leq \gamma(t) &\leq \tilde{\gamma}(t,y,c) + \frac{1}{2c}\max\{y^2,(1-y)^2\} \\ &\leq \tilde{\gamma}(t,y,c) + \frac{1}{2c} \qquad \forall t \in R. \end{aligned} \tag{8}$$

Thus the parameter c controls the accuracy of the approximation. The parameter y determines whether the approximation is more accurate for positive or negative values of the argument t. Thus for the extreme case $y = 1$ the approximation to $\gamma(t)$ is exact for $0 \leq t$ while for $y = 0$ the approximation is exact for $t \leq 0$.

Let us now formally describe the approximation procedure for solving problem (1) where we assume that the nondifferentiability of g is exclusively due to the presence of terms

$$\gamma[f_i(x)] = \max\{0,f_i(x)\} \quad , \ i \in I \tag{9}$$

and I is an arbitrary index set.

Given parameters c_k, y_k^i, $i \in I$ with $c_k > 0$, $0 \leq y_k^i \leq 1$, replace each term $\gamma[f_i(x)]$, $i \in I$ in the functional expression of g by $\tilde{\gamma}[f_i(x),y_k^i,c_k]$ to obtain a function $\tilde{g}_k$ and solve the problem

$$\begin{aligned} &\text{minimize } \tilde{g}_k(x) \\ &\text{subject to } x \in Q \subset X. \end{aligned} \tag{10}$$

If x_k is a solution of the above problem update c by setting $c_{k+1} = \beta c_k$, where $\beta > 1$, update the multipliers y_k^i in some fashion to obtain y_{k+1}^i, with $0 \leq y_{k+1}^i \leq 1$, $i \in I$, and solve the problem

$$\begin{aligned} &\text{minimize } \tilde{g}_{k+1}(x) \\ &\text{subject to } x \in Q \subset X \end{aligned} \tag{11}$$

to obtain a solution x_{k+1}.

It is important to note that the choice of the approximation (5) is by no means arbitrary and in fact it is closely related to penalty and multiplier methods for constrained minimization (see e.g. [8,3,4,11,16,18]). By introducing an auxiliary variable z, a simple kink may be written as

$$\gamma[f(x)] = \begin{aligned} &\min z \\ f(x) &\leq z \\ 0 &\leq z. \end{aligned} \tag{12}$$

By using a quadratic penalty function, the minimization problem above may be approximated by

$$\min_{0\leq z} \left[z + \frac{c}{2}\,(\max\{0,f(x)-z\})^2\right] \qquad c > 0. \tag{13}$$

Carrying out the minimization with respect to z, the expression above is equal to

$$\tilde{\gamma}[f(x),0,c] = \begin{cases} f(x) - \frac{1}{2c} & \text{if } \frac{1}{c} \leq f(x) \\ \frac{c}{2}[f(x)]^2 & \text{if } 0 \leq f(x) \leq \frac{1}{c} \\ 0 & \text{if } f(x) \leq 0 \end{cases} .$$

If we use the generalized quadratic penalty function used in the method of multipliers [4,16] the minimization problem in (12) may be approximated by the problem

$$\min_{0\leq z}\left[z + \frac{1}{2c}\left[(\max\{0,y + c[f(x)-z]\})^2 - y^2\right]\right] \quad , 0 < c, \quad 0 \leq y \leq 1. \tag{14}$$

Again by carrying out the minimization explicitly, the expression above is equal to $\tilde{\gamma}[f(x),y,c]$ as given by (5). Notice that we limit the range of the multiplier y to the interval [0,1] since one may prove that the Lagrange multiplier for problem (12) lies in that interval.

The interpretation of the approximation procedure in terms of penalty and multiplier methods is very valuable for a number of reasons. First it provides guidelines for approximation of simple kinks by a wide variety of functions. Every penalty function suitable for a penalty or multiplier method yields an approximating function via the procedure described above. A wide class of such functions is given in [13,14]. Many of these functions yield twice differentiable approximating functions, a property which may be desirable from the computational point of view. Second, our interpretation reveals that we may expect that the basic attributes of the behavior of penalty and multiplier methods will also be present in our approximation procedures. Thus we may expect ill-conditioning for large values of the parameter c. This fact necessitates sequential operation of the approximation method, i.e. repetitive solution of the approximate problem for ever increasing values of the parameter c. Third, the interpretation motivates us to consider an updating procedure for the parameters y which is analogous to the one used in the method of multipliers. This updating procedure results in most cases in significant improvements in computational efficiency.

The updating formula for the multipliers y_k^i, which closely parallels the one of the method of multipliers and which will be discussed in some detail in this paper, is given by

$$y_{k+1}^i = \begin{cases} 1 & \text{if } 1 \leq y_k^i + c_k f_i(x_k) \\ y_k^i + c_k f_i(x_k) & \text{if } 0 \leq y_k^i + c_k f_i(x_k) \leq 1 \\ 0 & \text{if } y_k^i + c_k f_i(x_k) \leq 0 \end{cases} . \tag{15}$$

One heuristic way to justify (15) is based on the following observation. If we assume for a moment that $x_k \to \bar{x}$ where $\bar{x}$ is an optimal solution of

(1) for some k we have $\frac{1-y_k^i}{c_k} \leq f_i(x_k)$ this indicates that likely there holds $0 \leq f_i(\bar{x})$ and therefore it is better to approximate more accurately the simple kink $\gamma[f_i(x)]$ for positive values of $f_i(x)$ rather than for negative values. The iteration (15) accomplishes precisely that by setting $y_{k+1}^i = 1$ (c.f. (5)). Similarly for the other cases $f_i(x_k) \leq -\frac{y_k^i}{c_k}$ and $-\frac{y_k^i}{c_k} \leq f_i(x_k) \leq \frac{1-y_k^i}{c_k}$ the iteration (15) may be viewed as adaptively adjusting the accuracy of approximation of the simple kink $\gamma(t)$ from positive to negative values of the scalar argument t and vice versa as the circumstances dictate. A more rigorous and satisfying justification for the employment of (15) together with a clarification of the connection with the method of multipliers is provided in [21].

Since the class of problems of the form (1) is extremely rich in variety, it is very difficult to provide a convergence and rate of convergence analysis for the general case. The utter impossibility of providing a unifying notational description for the general case of problem (1) is one of the main obstacles here. For this reason we shall restrict ourselves to the specific class of problems

$$\begin{aligned} &\text{minimize } g[x,\gamma[f_1(x)],\ldots,\gamma[f_m(x)]] \\ &\qquad\text{subject to } x \in Q \subset R^n \end{aligned} \tag{16}$$

where g, $f_1,\ldots,f_m$ are continuously differentiable functions. Our results also apply with simple modifications in statement and proof to the class of problems

$$\begin{aligned} &\text{minimize } g[x,\max\{f_1^1(x),\ldots,f_1^{s_1}(x)\},\ldots,\max\{f_m^1(x),\ldots,f_m^{s_m}(x)\}] \\ &\qquad\text{subject to } x \in Q \subset R. \end{aligned} \tag{17}$$

On the other hand our analysis can serve as the prototype for the analysis of other different or more general cases and provides a measure of what kind of behavior one may expect from the approximation methods that we propose.

The paper is organized as follows: In the next section we prove the basic convergence results for our approximation methods. Section 3 shows that one may obtain, as a byproduct of the computation, quantities which play a role analogous to Lagrange multipliers in constrained minimization. We also show that our convergence results may be used to obtain some optimality conditions for the problem that we are considering. Section 4 examines the possibility of acceleration of convergence by using iteration (15). The connection with the method of multipliers is clarified and some convergence and rate of convergence results are inferred. Finally in Section 5 we present some computational results.

2. SOME CONVERGENCE RESULTS

Consider problem (16) where we adopt the standing assumption that the set Q is nonempty and that the functions $g:R^{n+m} \to R$, $f_1:R^n \to R$, $i = 1,\ldots,m$ are everywhere continuously differentiable. We denote by $\nabla_x g$ the (column) vector of the first n partial derivatives of g, while we denote by $\frac{\partial g}{\partial t_i}$, $i = 1,\dots,m$ the partial derivative of g with respect to the (n+i)th argument (i.e. $\frac{\partial g}{\partial t_i} = \frac{\partial g}{\partial \gamma[f_i(x)]}$).

Consider now the kth approximation minimization problem

$$\min_{x\in Q}\ g[x,\tilde{\gamma}[f_1(x),y_k^1,c_k],\dots,\tilde{\gamma}[f_m(x),y_k^m,c_k]] \tag{18}$$

where

$$0 < c_k < c_{k+1},\quad c_k \to \infty,\quad 0 \le y_k^i \le 1,\quad i = 1,\dots,m,\quad k = 0,1,\dots$$

and the approximate kink $\tilde{\gamma}[f_i(x),y_k^i,c_k]$ is given by (5). Any rule may be used for updating y_k^i - for example y_k^i may be left constant. Let x_k be an optimal solution of problem (18) (assuming one exists). We have the following basic proposition:

Proposition 1: There holds

$$|g^* - g[x_k,\gamma[f_1(x_k)],\dots,\gamma[f_m(x_k)]]| \le \frac{L}{c_k} \qquad k = 0,1,\dots \tag{19}$$

where

$$g^* = \inf_{x\in Q} g[x,\gamma[f_1(x)],\dots,\gamma[f_m(x)]] \tag{20}$$

$$L = \sum_{i=1}^{m} \sup_{(x,t_1,\dots,t_m)\in M} \left|\frac{\partial g}{\partial t_i}(x,t_1,\dots,t_m)\right| \tag{21}$$

$$M = \{(x,t_1,\dots,t_m) \mid x\in Q,\ \gamma[f_1(x)] - \frac{1}{2c_o} \le t_i \le \gamma[f_1(x)],\ i = 1,\dots,m\} \tag{22}$$

provided L above is finite.

Proof: By Taylor's theorem, (8) and (21), (22) we have for every $x\in Q$

$$|g[x,\gamma[f_1(x)],\dots,\gamma[f_m(x)]] - g[x,\tilde{\gamma}[f_1(x),y_k^1,c_k],\dots,\tilde{\gamma}[f_m(x),y_k^m,c_k]]| \le \frac{L}{2c_k} \tag{23}$$

from which the result follows. Q.E.D.

As a direct consequence of the proposition above we have the following convergence results:

Corollary 1.1: Let Q be a bounded set. Then

$$\lim_{k\to\infty} g[x_k,\gamma[f_1(x_k)],\dots,\gamma[f_m(x_k)]] = g^*.$$

Proof: The boundedness of Q implies that L as given by (21) is finite. Hence by (19) the fact $c_k \to \infty$ the result follows. Q.E.D.

Corollary 1.2: Let g have the particular form

$$g[x,\gamma[f_1(x)],\dots,\gamma[f_m(x)]] = g_o(x) + \sum_{i=1}^{m} \gamma[f_i(x)] \qquad , g_o : R^n \to R.$$

Then

$$|g^* - g[x_k,\gamma[f_1(x_k)],\dots,\gamma[f_m(x_k)]]| \le \frac{m}{c_k}.$$

Proof: Immediate from (23), (21), (19). Q.E.D.

The above corollary is interesting from a computational point of view since it shows that for the problem above there are available a priori bounds on the approximation error.

Corollary 1.3: Let $\bar{x}$ be any limit point of the sequence $\{x_k\}$ and let Q be a closed set. Then

$$g[\bar{x},\gamma[f_1(\bar{x})],\ldots,\gamma[f_m(\bar{x})]] = \min_{x\in Q} g[x,\gamma[f_1(x)],\ldots,\gamma[f_m(x)]]$$

i.e. $\bar{x}$ is an optimal solution of problem (16).

Proof: Without loss of generality assume that the whole sequence $\{x_k\}$ converges to $\bar{x}$ and let S be any closed sphere containing the sequence $\{x_k\}$. Clearly each vector x_k is an optimal solution of the problem

$$\min_{x\in Q\cap S} g[x,\tilde{\gamma}[f_1(x),y_k^1,c_k],\ldots,\tilde{\gamma}[f_m(x),y_k^m,c_k]]$$

and since $Q\cap S$ is bounded, by Corollary 1.1 and (23) we have

$$\lim_{k\to\infty} g[x_k,\tilde{\gamma}[f_1(x_k),y_k^1,c_k],\ldots,\tilde{\gamma}[f_m(x_k),y_k^m,c_k]] = g[\bar{x},\gamma[f_1(\bar{x})],\ldots,$$
$$\gamma[f_m(\bar{x})]] = \min_{x\in Q\cap S} g[x,\gamma[f_1(x)],\ldots,\gamma[f_m(x)]].$$

It remains to show that the minimum of g over $Q\cap S$ above is equal to the minimum of g over Q. But this follows from the fact that S can be any closed sphere containing $\{x_k\}$ and hence it can have an arbitrarily large radius. Q.E.D.

Notice that the proposition states that every limit point of $\{x_k\}$ is a minimizing point of problem (16), but does not guarantee the existence of at least one limit point. This is to be expected since problem (16) may not have a solution. On the other hand the existence of a solution as well as of at least one limit point of $\{x_k\}$ is guaranteed when Q is a compact set.

The above proposition and corollaries establish the validity of the approximation procedure. One may notice that the proofs are very simple and rest on the fact that the function $\tilde{\gamma}[f_i(x),y_k^i,c_k]$ approximates uniformly the simple kink $\gamma[f_i(x)]$ with an approximation error at most equal to $\frac{1}{2c_k}$ as shown by (8). It is interesting to observe that convergence does not depend on the particular values y_k^i, $i\in I$ employed. This allows a great deal of freedom in adjusting y_k^i for the purpose of accelerating convergence.

Since our procedures are related to penalty methods one expects that they must yield, as a by-product of the computation, quantities which may be viewed as Lagrange multipliers. In the next section we show that such multipliers may indeed be obtained. Furthermore we show in [21] that these multipliers enter in optimality conditions which, aside from their analytical value, may serve as a basis for termination of our approximation procedure.

3. MULTIPLIER CONVERGENCE AND CONDITIONS FOR OPTIMALITY

Let us assume throughout this section that Q is a closed convex set. Using (9), the gradient with respect to x of the objective function in problem (18) may be calculated to be

$$\nabla g[x,\tilde{\gamma}[f_1(x),y_k^1,c_k],\dots,\tilde{\gamma}[f_m(x),y_k^m,c_k]] = \nabla_x g + \sum_{i=1}^{m} \frac{\partial g}{\partial t_i} \nabla\tilde{\gamma}[f_i(x),y_k^i,c_k]\Big|_x$$

$$= \nabla_x g + \sum_{i=1}^{m} \frac{\partial g}{\partial t_i} \tilde{y}_k^i(x)\nabla f_i\Big|_x \tag{25}$$

where $\tilde{y}_k^i$, $i=1,\dots,m$ is given by

$$\tilde{y}_k^i(x) = \begin{cases} 1 & \text{if } \ 1 \le y_k^i + c_k f_i(x) \\ y_k^i + c_k f_i(x) & \text{if } \ 0 \le y_k^i + c_k f_i(x) \le 1 \\ 0 & \text{if } \ y_k^i + c_k f_i(x) \le 0 \end{cases} \tag{26}$$

and all gradients in the right hand side of (25) are evaluated at the point x.

Since Q is a convex set and x_k is an optimal solution of problem (18) we have the necessary condition

$$\langle \nabla_x g + \sum_{i=1}^{m} \frac{\partial g}{\partial t_i} \tilde{y}_k^i(x_k)\nabla f_i\Big|_{x=x_k} , \ (x-x_k)\rangle \ge 0 \qquad \forall x \in Q$$

where $\langle\cdot,\cdot\rangle$ denotes the usual inner product in R^n.

Let now $\{x_k\}_{k\in K}$ be a subsequence of $\{x_k\}$ which converges to $\bar{x}$. By Corollary 1.3, $\bar{x}$ is an optimal solution of problem (16). In addition the subsequence $\{\tilde{y}_k\} = \{\tilde{y}_k^1(x_k),\dots,\tilde{y}_k^m(x_k)\}_{k\in K}$ defined by (26) has at least one limit point, and, by taking limits in (27), each of its limit points $\bar{y} = \{\bar{y}^1,\dots,\bar{y}^m\}$ must satisfy

$$\langle \nabla_x g + \sum_{i=1}^{m} \frac{\partial g}{\partial t_i} \bar{y}^i \nabla f_i\Big|_{x=\bar{x}} , \ (x-\bar{x})\rangle \ge 0 \qquad \forall x \in Q.$$

Combining the above observations with Corollary 1.3 we obtain the following proposition:

Proposition 2: Let Q be a closed convex set, and let $(\bar{x},\bar{y})$ be any limit point of the sequence $\{x_k,\tilde{y}_k\}$, where $\tilde{y}_k = (\tilde{y}_k^1(x_k),\dots,\tilde{y}_k^m(x_k))$ is defined by (26). Then $\bar{x}$ is an optimal solution of problem (16), and $\bar{y}$ is a multiplier vector satisfying

$$\langle \nabla_x g + \sum_{i=1}^{m} \frac{\partial g}{\partial t_i} \bar{y}^i \nabla f_i\Big|_{x=\bar{x}} , \ (x-\bar{x})\rangle \ge 0 \qquad \forall x \in Q \tag{28}$$

$$\bar{y}^i = 0 \ \text{ if } f_i(\bar{x}) < 0, \quad \bar{y}^i = 1 \ \text{ if } f(\bar{x}) > 0, \quad 0 \le \bar{y}^i \le 1 \ \text{ if } f_i(\bar{x}) = 0.$$

Proposition 2 together with Proposition 1 and its corollaries may be used to provide a simple proof of an optimality condition for problem (16) which is given in [21].

4. ACCELERATION BY MULTIPLIER ITERATIONS

In this section we mention a device which may be used to improve the computational efficiency of the approximation method. We consider the case where the approximation procedure is operated sequentially and the multipliers y_k^i, $i \in I$ used in the approximations, are updated by means of the iteration

$$y_{k+1}^i = \begin{cases} 1 & \text{if } 1 \leq y_k^i + c_k f_i(x_k) \\ y_k^i + c_k f_i(x_k) & \text{if } 0 \leq y_k^i + c_k f_i(x_k) \leq 1 \\ 0 & \text{if } y_k^i + c_k f_i(x_k) \leq 0. \end{cases} \tag{29}$$

A heuristic justification for this iteration was given in the introduction where we also mentioned its connection with iterations for the method of multipliers. This connection is discussed in some detail in [21]. In our computational experiments the employment of iteration (29) speeded up convergence a great deal.

Finally we note that when the constraint set Q in problem (16) is specified by equality and inequality constraints it is convenient to eliminate these constraints by means of penalty functions while solving the approximate minimization problems. In this way the approximation method is combined with the penalty function method in a natural way. One may use the same parameter c_k to control both the accuracy of the approximation and the severity of the penalty. Assuming that $c_k \to \infty$ one may prove various convergence theorems for methods of this type simply by combining standard arguments of convergence proofs of penalty methods [8] together with the convergence arguments of this paper. As an example consider the problem

$$\text{minimize } g_o[x, \gamma[f_1(x)], \ldots, \gamma[f_m(x)]]$$

$$\text{subject to } g_1[x, \gamma[h_1(x)], \ldots, \gamma[h_p(x)]] = 0$$

where g_o, g_1 are real valued functions. One may consider sequential minimization of the function

$$\begin{aligned} & g_o[x, \tilde{\gamma}[f_1(x), y_k^1, c_k], \ldots, \tilde{\gamma}[f_m(x), y_k^m, c_k]] + \\ & + \lambda_k g_1[x, \tilde{\gamma}[h_1(x), w_k^1, c_k], \ldots, \tilde{\gamma}[h_p(x), w_k^p, c_k]] + \\ & + \frac{c_k}{2} [g_1[x, \tilde{\gamma}[h_1(x), w_k^1, c_k], \ldots, \tilde{\gamma}[h_p(x), w_k^p, c_k]]]^2 \end{aligned}$$

where $\{c_k\}$ is an increasing sequence tending to infinity with $c_k > 0$, $\{y_k^i\}$, $\{w_k^i\}$ satisfy

$$0 \leq y_k^i \leq 1 \quad , \ 0 \leq w_k^i \leq 1 \qquad \text{for all } k, i$$

and $\{\lambda_k\}$ is a bounded scalar sequence. The updating procedure for the multipliers which corresponds to the iteration of the method of multipliers is given by

$$y_{k+1}^i = \begin{cases} 1 & \text{if } 1 \leq y_k^i + c_k f_i(x_k) \\ y_k^i + c_k f_i(x_k) & \text{if } 0 \leq y_k^i + c_k f_i(x_k) \leq 1 \\ 0 & \text{if } y_k^i + c_k f_i(x_k) \leq 0 \end{cases} \quad (30)$$

$$w_{k+1}^i = \begin{cases} 1 & \text{if } 1 \leq w_k^i + c_k h_i(x_k) \\ w_k^i + c_k h_i(x_k) & \text{if } 0 \leq w_k^i + c_k h_i(x_k) \leq 1 \\ 0 & \text{if } y_k^i + c_k h_i(x_k) \leq 0 \end{cases} \quad (31)$$

$$\lambda_{k+1} = \lambda_k + c_k g_i[x_k, \tilde{\gamma}[h_1(x_k), w_k^1, c_k], \ldots, \tilde{\gamma}[h_p(x_k), w_k^p, c_k]]. \quad (32)$$

The updating procedure (35)-(37) appears to be a reasonable one and when employed it improved a great deal the speed of convergence in our computational experiments. However we offer here no theoretical analysis which supports the conjecture that it accelerates convergence for any broad class of problems.

5. COMPUTATIONAL RESULTS

We have performed, with the assistance of L. Berman, a number of computational experiments in order to test the analysis of this paper. We summarize here some partial results for the unconstrained problem

$$\underset{x}{\text{minimize}} \; (1 + \sum_{i=1}^{n} i|x^i|)^2. \quad (33)$$

The following constrained version of the problem was also considered

$$\text{minimize} \; (1 + \sum_{i=1}^{n} i|x^i|)^2$$
$$\text{subject to } |x^1 - 2| + |x^2| + \cdots + |x^n| = 1. \quad (34)$$

We obtained the optimal value of problem (33) within six significant digits of accuracy by employing the method described in Section 2 where we either kept the multipliers y_k^i fixed to zero or updated them by means of the iteration (29). For $n = 5$ and $c_k = 5^k$ the method without updating y_k^i required 188 iterations of the Fletcher-Powell method that we employed and when y_k^i were updated only 41 iterations were required. For $n = 50$ and $c_k = 5^k$ the corresponding number of iterations was 1532 and 378. We also tried a method where c_k was kept fixed at the value 10 ($c_k = 10, \forall k$) while the multipliers were updated via (29). The method converged with no difficulty and the required number of iterations was 26 for $n = 5$ and 498 for $n = 50$.

The constrained problem (34) was solved by the combination of the method of multipliers and the approximation method described at the end of the previous section. For $n = 5$ and $c_k = 5^k$ the method with all multipliers fixed to zero required 224 iterations while when the multipliers were updated by means of (30), (31), (32) the method required 50 iterations. For $n = 50$ and $c_k = 5^k$ the number of iterations required was 2,627 and 630 respectively. We also kept c_k fixed at the value 10 and updated the multipliers via (30), (31), (32) and the method performed successfully, The number of iterations required was 61 (for n = 5) and 856 (for n = 50).

REFERENCES

[1] Bertsekas, D. P. and Mitter, S. K., "A Descent Numerical Method for Optimization Problems with Nondifferentiable Cost Functionals," SIAM J. Control, Vol. 11, 1973, pp. 637-652.

[2] Bertsekas, D. P. and Mitter, S. K., "Steepest Descent for Optimization Problems with Nondifferentiable Cost Functionals," Proc. of 5th Annual Princeton Conference on Information Sciences and Systems, Princeton, N. J., March 1971.

[3] Bertsekas, D. P., "Combined Primal-Dual and Penalty Methods for Constrained Minimization," EES Dept. Working Paper, Stanford Univ., Stanford, Calif., Jan. 1973, to appear in SIAM J. Control, Aug. 1975.

[4] Bertsekas, D. P., "On Penalty and Multiplier Methods for Constrained Minimization," EES Dept. Working Paper, Stanford Univ., Stanford, Calif., August 1973, to appear in SIAM J. on Control.

[5] Cullum, J., Donath, W. E., and Wolfe, P., "An Algorithm for Minimizing Certain Nondifferentiable Convex Functions," RC 4611, IBM Research, Yorktown Heights, N. Y., November 1973.

[6] Demyanov, V. F., "On the Solution of Certain Minimax Problems," Kibernetica, Vol. 2, 1966.

[7] Demyanov, V. F. and Rubinov, A. M., Approximate Methods in Optimization Problems, American Elsevier Publ. Co., New York, 1970.

[8] Fiacco, A. V. and McCormick, G. P., Nonlinear Programming: Sequential Unconstrained Minimization Techniques, J. Wiley, New York, 1968.

[9] Goldstein, A. A., Constructive Real Analysis, Harper & Row, New York, 1967.

[10] Held, M., Wolfe, P., and Crowder, H. P., "Validation of Subgradient Optimization," Mathematical Programming, Vol. 6, No. 1, 1974, pp. 62-88.

[11] Hestenes, M. R., "Multiplier and Gradient Methods," Journal of Optimization Theory and Applications, Vol. 4, No. 5, 1969, pp. 303-320.

[12] Lemarechal, C., "An Algorithm for Minimizing Convex Functions," Proc. IFIP Congress 74, August 1974.

[13] Kort, B. W. and Bertsekas, D. P., "Combined Primal-Dual and Penalty Methods for Convex Programming," EES Dept. Working Paper, Stanford Univ., Stanford, Calif., August 1973, submitted for publication.

[14] Kort, B. W. and Bertsekas, D. P., "Multiplier Methods for Convex Programming," Proc. of 1973 IEEE Conference on Decision and Control, San Diego, Calif., December 1973, pp. 428-432.

[15] Polyak, B. T., "Minimization of Unsmooth Functionals," Zh. Vychisl. Mat. i Mat. Fiz., Vol. 9, No. 3, 1969, pp. 509-521.

[16] Powell, M. J. D., "A Method for Nonlinear Constraints in Minimization Problems," in Optimization, R. Fletcher (ed.), Academic Press, New York, 1969, pp. 283-298.

[17] Pschenishnyi, B. N., "Dual Methods in Extremum Problems," Kibernetica, Vol. 1, No. 3, 1965, pp. 89-95.

[18] Rockafellar, R. T., "The Multiplier Method of Hestenes and Powell Applied to Convex Programming," J. Opt. Theory Appl., Vol. 12, No. 6, 1973.

[19] Shor, N. Z., "On the Structure of Algorithms for the Numerical Solution of Planning and Design Problems," Dissertation, Kiev, 1964.

[20] Wolfe, P., "A Method of Conjugate Subgradients for Minimizing Nondifferentiable Functions," RC 4611, IBM Research, Yorktown Heights, N. Y., May 1974.

[21] Bertsekas, D. P., "Nondifferentiable Optimization via Approximation," Coordinated Science Lab. Working paper, September 1974, submitted for publication.

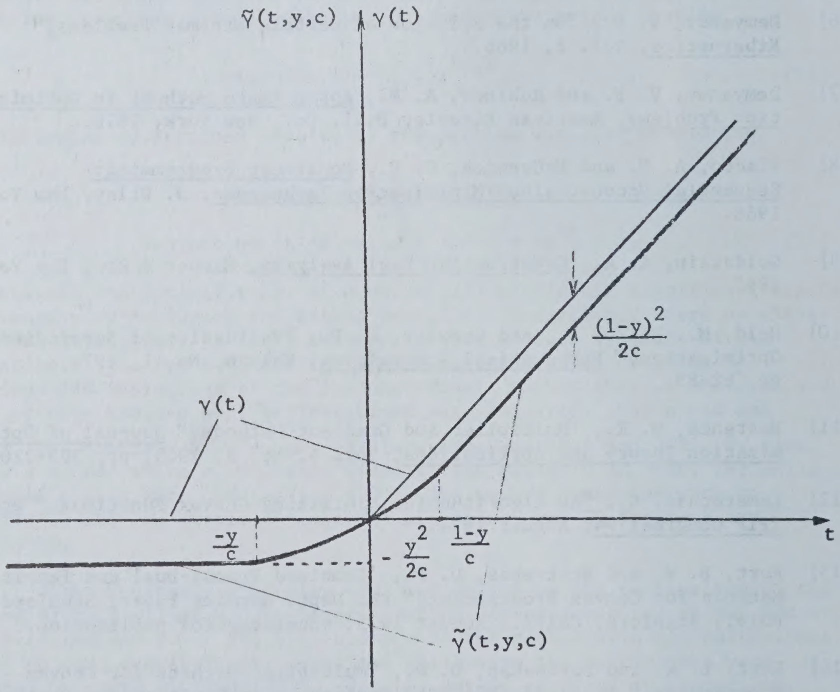


Figure 1

THE STABILIZABILITY OF LARGE INTERCONNECTED SYSTEMS

EDWARD J. DAVISON*
Department of Electrical Engineering
University of Toronto, Canada

ABSTRACT

Using Rosenbrock's [1] frequency domain approach [1], necessary and sufficient conditions are found for a linear time-invariant system, consisting of a number of sub-systems interconnected in a very general manner, which includes series, parallel, feedback and hierarchial systems, to be stabilizable. It is then shown that, under certain mild conditions, almost all interconnected systems are stabilizable.

I. INTRODUCTION

There has been much interest recently in determining conditions which guarantee that a composite system is either controllable or observable in terms of the parameters of the subsystems. See for example [1, p.125]-[8]. The present development obtains necessary and sufficient conditions for a general series-parallel or feedback composite system, which includes hierarchial systems as a special case, to be either stabilizable (controllable) or detectable (observable) in terms of the state equations of the composite system. The conditions obtained are very simple to compute.

Part 2 of the paper deals with the special case when the composite system consists of only 2 subsystems and part 3 of the paper deals with the general case when the composite system consists of γ subsystems.

II. PRELIMINARY RESULTS

The following definitions will be used in the development to follow.

Definition

(A,B) is said to be stabilizable, if there exists a feedback map K so that A+BK is stable. Let a coordinate transformation T be applied to (C,A) so that $(CT, T^{-1}AT) \sim \left\{(C_1, 0), \begin{bmatrix} A_1 & 0 \\ A_3 & A_2 \end{bmatrix}\right\}$ where (C_1, A_1) is observable; then (C, A) is said to be detectable if A_2 is stable.

Remark

1. It is clear that if (C, A, B) is stabilizable and detectable, then the resultant closed loop system can be stabilized by using only output feedback, e.g. by using an observer.

---

* Department of Electrical Engineering, University of Toronto, Toronto, Canada. This work has been supported by the National Research Council of Canada under Grant No. A4396.

Series-Parallel Composite Systems

Given the general series-parallel composite system:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} A_1 & E \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u$$

$$y = (C_1 \quad C_2) \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + Du \tag{1}$$

with input u and output y, where $x_1 \varepsilon R^{n_1}$, $x_2 \varepsilon R^{n_2}$, $u \varepsilon R^m$, $y \varepsilon R^r$, let the eigenvalues of A_1 be given by λ_i^1, $i = 1,2,\ldots,n_1$ and let the eigenvalues of A_2 be given by λ_i^2, $i = 1,2,\ldots,n_2$. Let the unstable eigenvalues of A_1 and A_2 be given by λ^{*1}_i, $i = 1,2,\ldots,n_1^*$ and λ^{*2}_i, $i = 1,2,\ldots,n_2^*$ respectively. The following result is obtained:

Theorem 1

The series-parallel composite system (1) is stabilizable if and only if the following two conditions both hold:

$$\text{(a)} \quad \text{rank} \ (A_2 - \lambda^{*2}_i I, \ B_2) = n_2 \ , \ i = 1,2,\ldots,n_2^* \tag{2}$$

$$\text{(b)} \quad \text{rank} \begin{bmatrix} A_1 - \lambda^{*1}_i I & E & B_1 \\ 0 & A_2 - \lambda^{*1}_i I & B_2 \end{bmatrix} = n_1 + n_2 \ , \ i = 1,2,\ldots,n_1^* \tag{3}$$

and the system (1) is detectable if and only if the following two conditions both hold:

$$\text{(c)} \quad \text{rank} \begin{bmatrix} A_1 - \lambda^{*1}_i I \\ C_1 \end{bmatrix} = n_1 \quad , \quad i = 1,2,\ldots,n_1^* \tag{4}$$

$$\text{(d)} \quad \text{rank} \begin{bmatrix} A_1 - \lambda^{*2}_i I & E \\ 0 & A_2 - \lambda^{*2}_i I \\ C_1 & C_2 \end{bmatrix} = n_1 + n_2 \ , \ i = 1,2,\ldots,n_2^* \tag{5}$$

Proof

Due to page restrictions, all proofs of all results are omitted; they may be found in [9].

Feedback Composite Systems

Given the feedback composite system:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} A_1 & -B_1C_2 \\ B_2C_1 & A_2 - B_2D_1C_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2D_1 \end{bmatrix} u$$

$$y = (C_1 \ , \ -D_1C_2) \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + D_1 \ u \tag{6}$$

with input u and output y, where $x_1 \varepsilon R^{n_1}$, $x_2 \varepsilon R^{n_2}$, $u \varepsilon R^m$, $y \varepsilon R^r$, let the eigenvalues of A_1 and A_2 be denoted by the same values as in (1).

The above composite system arises from the following feedback system:

$$\dot{x}_1 = A_1x_1 + B_1e \quad , \quad \dot{x}_2 = A_2x_2 + B_2y \quad , \quad e = u - z \qquad (7)$$
$$y = C_1x_1 + D_1e \qquad\qquad z = C_2x_2$$

The following result is obtained:

Theorem 2

The feedback composite system (6) is stabilizable if and only if the following two conditions both hold:

$$\text{(a)} \quad \text{rank } (A_1 - \overset{*}{\lambda}{}_i^1 I,\ B_1) = n_1 \quad , \quad i = 1,2,\ldots,\overset{*}{n}_1 \qquad (8)$$

$$\text{(b)} \quad \text{rank} \begin{bmatrix} A_1 - \overset{*}{\lambda}{}_i^2 I & 0 & B_1 \\ B_2C_1 & A_2 - \overset{*}{\lambda}{}_i^2 I & B_2D_1 \end{bmatrix} = n_1 + n_2 \ , \ i=1,2,\ldots,\overset{*}{n}_2 \qquad (9)$$

and the system (6) is detectable if and only if the following two conditions both hold:

$$\text{(c)} \quad \text{rank} \begin{bmatrix} A_1 - \overset{*}{\lambda}{}_i^1 I \\ C_1 \end{bmatrix} = n_1 \ , \ i = 1,2,\ldots,\overset{*}{n}_1 \qquad (10)$$

$$\text{(d)} \quad \text{rank} \begin{bmatrix} A_1 - \overset{*}{\lambda}{}_i^2 I & -B_1C_2 \\ 0 & A_2 - \overset{*}{\lambda}{}_i^2 I \\ C_1 & -D_1C_2 \end{bmatrix} = n_1 + n_2, \ i = 1,2,\ldots,\overset{*}{n}_2 \qquad (11)$$

Remarks

2. The above two theorems are valid if "stabilizable" is replaced by "controllable" and if "detectable" is replaced by "observable", provided $\overset{*}{\lambda}{}_i^2$, $\overset{*}{n}_2$ are replaced by λ_i^2 , n_2 respectively and $\overset{*}{\lambda}{}_i^1$, $\overset{*}{n}_1$ are replaced by λ_i^1 , n_1 respectively.

3. The composite systems (1), (6) considered, include as special cases all other previous results treated before in [1] - [7].

4. It should be noted that the conditions obtained in Theorems 1,2 are remarkably easy to check, it just being a matter of determining eigenvalues & finding the rank of a small dimension matrix. In this regard, the new conditions obtained should be much more advantageous to use computationally then the standard conditions obtained by decomposing the system into its controllable and uncontrollable parts. This is because the new conditions do not require the calculation of A^jB , $j = 0,1,2,\ldots$ terms, which in the case of high order systems, are known to produce numerical difficulties due to the finite word length of a digitalcomputer.

The following theorems show that a general series-parallel feedback cascade system is stabilizable or/and detectable for "almost all" [10] inter-connection matrices, provided some mild conditions on the subsystems are imposed.

Definition

If $f = f(x_1,\dots,x_n)$ is a polynomial in the n variables $x_1,\dots,x_n$ with coefficients in R, then the point $\hat{x} = (\hat{x}_1,\dots,\hat{x}_n)$ in R^n is said to be a zero of f if $f(\hat{x}_1,\dots,\hat{x}_n) = 0$. The set of zeros of f is called the locus of f, and is denoted by V(f) . A proper subset V of R^n is called a *hypersurface* in R^n if it is the locus of a nonconstant polynomial.

Theorem 3

Assume that $\left\{(C_1,C_2)\ ,\begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix},\ \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}\right\}$ is stabilizable and detectable; then

(a) The class of all interconnection matrices E which makes (1) not both stabilizable and detectable is either an empty set or a subset of a hypersurface in the parameter space of E.

(b) The class of all interconnection matrices (E_1,E_2) which makes $\left\{(C_1,C_2)\ ,\ \begin{bmatrix} A_1 & E_1 \\ E_2 & A_2 \end{bmatrix},\ \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}\right\}$ not both stabilizable and detectable is either an empty set or a subset of a hypersurface in the parameter space of (E_1,E_2).

Remark 5

The requirement that $\left\{(C_1,C_2)\ ,\begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix},\ \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}\right\}$ be stabilizable and detectable is a fairly strong condition; a weaker condition is assumed in the following result.

Theorem 4

Assume that (A_2,B_2) is stabilizable, that (C_1,A_1) is detectable and that A_1,A_2 each have distinct eigenvalues; then:

(a) The class of all interconnection matrices E which makes $\left\{(C_1,0)\ ,\ \begin{bmatrix} A_1 & E \\ 0 & A_2 \end{bmatrix},\begin{bmatrix} 0 \\ B_2 \end{bmatrix}\right\}$ not both stabilizable and detectable is either an empty set or a subset of a hypersurface in the parameter space of E.

(b) The class of all matrices (E,C_2,B_1) which makes $\left\{(C_1,C_2)\ ,\begin{bmatrix} A_1 & E \\ 0 & A_2 \end{bmatrix},\begin{bmatrix} B_1 \\ B_2 \end{bmatrix}\right\}$ not both stabilizable and detectable is either an empty set or a subset of a hypersurface in the parameter space of (E,C_2,B_1).

(c) The class of all interconnection matrices (E_1,E_2) which makes $\left\{(C_1,0)\ ,\ \begin{bmatrix} A_1 & E_1 \\ E_2 & A_2 \end{bmatrix},\ \begin{bmatrix} 0 \\ B_2 \end{bmatrix}\right\}$ not both stabilizable and detectable is either an empty set or a subset of a hypersurface in the parameter space of (E_1, E_2).

(d) The class of all matrices (E_1,E_2,B_1,C_2) which makes $\left\{(C_1,C_2)\ ,\begin{bmatrix} A_1 & E_1 \\ E_2 & A_2 \end{bmatrix}\ ,\ \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}\right\}$ not both stabilizable and detectable is either an empty set or a subset of a hypersurface in the parameter space of (E_1,E_2,B_1,C_2).

Remarks

6. Theorem 4 is most interesting; it implies, for example, that if there exist two subsystems with distinct eigenvalues, then if one subsystem is stabilizable, almost all interconnection matrices between the two subsystems will result in the composite system being stabilizable.

7. The following example: $A_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$, $A_2 = (1)$, $B_2 = (1)$, $C_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ shows that Theorem 4 is not necessarily true unless A_1, A_2 each have distinct eigenvalues.

III. MAIN RESULTS : MULTI-COMPOSITE SYSTEMS

Given the general series-parallel multi-composite system:

$$\dot{x} = \begin{bmatrix} A_1 & E_{11} & E_{12} & E_{13} & \cdots & E_{1,\gamma-1} \\ 0 & A_2 & E_{22} & E_{23} & \cdots & E_{2,\gamma-1} \\ 0 & 0 & A_3 & E_{33} & \cdots & E_{3,\gamma-1} \\ 0 & 0 & 0 & A_4 & \cdots & E_{4,\gamma-1} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & E_{\gamma-1,\gamma-1} \\ 0 & 0 & 0 & 0 & \cdots & A_\gamma \end{bmatrix} x + \begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ B_4 \\ \vdots \\ B_{\gamma-1} \\ B_\gamma \end{bmatrix} u \tag{12}$$

$$y = (C_1 \quad C_2 \quad C_3 \quad C_4 \quad \cdots \quad C_\gamma)x + Du$$

with input u and output y, where $x_i \in R^{n_i}$, $i=1,2,\ldots,\gamma$, $u \in R^m$, $y \in R^r$, let the eigenvalues of A_i, $i=1,2,\ldots,\gamma$ be given by λ^i_j, $j=1,2,\ldots,n_i$; $i=1,2,\ldots,\gamma$ and let the unstable eigenvalues of A_i, $i=1,2,\ldots,\gamma$ be given by λ^{*i}_j , $j=1,2,\ldots,n^*_i$, $i=1,2,\ldots,\gamma$.

Remark 8

The above composite system is very general - it includes general series-parallel systems and hierarchial multi-level systems; for example, the following is a description of a 3 level hierarchial system:

$$
\begin{bmatrix} \dot{x}_7 \\ \dot{x}_6 \\ \dot{x}_5 \\ \dot{x}_4 \\ \dot{x}_3 \\ \dot{x}_2 \\ \dot{x}_1 \end{bmatrix}
=
\begin{bmatrix}
A_7 & 0 & 0 & 0 & 0 & x & 0 \\
0 & A_6 & 0 & 0 & 0 & x & 0 \\
0 & 0 & A_5 & 0 & x & 0 & 0 \\
0 & 0 & 0 & A_4 & x & 0 & 0 \\
0 & 0 & 0 & 0 & A_3 & 0 & x \\
0 & 0 & 0 & 0 & 0 & A_2 & x \\
0 & 0 & 0 & 0 & 0 & 0 & A_1
\end{bmatrix}
\begin{bmatrix} x_7 \\ x_6 \\ x_5 \\ x_4 \\ x_3 \\ x_2 \\ x_1 \end{bmatrix}
+
\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ B \end{bmatrix} u
$$

$$
y = (x \quad x \quad x \quad x \quad 0 \quad 0 \quad 0)
\begin{bmatrix} x_7 \\ x_6 \\ x_5 \\ x_4 \\ x_3 \\ x_2 \\ x_1 \end{bmatrix}
+ (x) \; u \qquad (13)
$$

where x denotes an element which is not necessarily zero, and where subsystem $\dot{x}_1 = A_1 x_1 + Bu$ is on the 1st level etc.

The following result is immediately obtained by applying Theorem 1 to (12) in a sequential manner:

Theorem 5

The series-parallel multi-composite system (12) is stabilizable if and only if the following conditions all hold:

(C1) $\operatorname{rank}\,(A_\gamma - \lambda_j^{*\gamma} I,\ B_\gamma) = n_\gamma \;, \quad j = 1,2,\ldots,n_\gamma^*$

(C2) $$\operatorname{rank}\begin{bmatrix} A_{\gamma-1} - \lambda_j^{*\gamma-1} I & E_{\gamma-1,\gamma-1} & B_{\gamma-1} \\ 0 & A_\gamma - \lambda_j^{*\gamma-1} I & B_\gamma \end{bmatrix} = n_{\gamma-1} + n_\gamma \;, \quad j = 1,2,\ldots,n_{\gamma-1}^*$$

(C3) $$\operatorname{rank}\begin{bmatrix} A_{\gamma-2} - \lambda_j^{*\gamma-2} I & E_{\gamma-2,\gamma-2} & E_{\gamma-2,\gamma-1} & B_{\gamma-2} \\ 0 & A_{\gamma-1} - \lambda_j^{*\gamma-2} I & E_{\gamma-1,\gamma-1} & B_{\gamma-1} \\ 0 & 0 & A_\gamma - \lambda_j^{*\gamma-2} I & B_\gamma \end{bmatrix} = n_{\gamma-2} + n_{\gamma-1} + n_\gamma \;, \quad j=1,2,\ldots,n_{\gamma-2}^* \qquad (13)$$

$$\vdots \qquad\qquad \vdots \qquad\qquad \vdots$$

(C_γ) $$\operatorname{rank}\begin{bmatrix}
A_1 - \lambda_j^{*1} I & E_{11} & E_{12} & E_{13} & \cdots & E_{1,\gamma-1} & B_1 \\
0 & A_2 - \lambda_j^{*1} I & E_{22} & E_{23} & \cdots & E_{2,\gamma-1} & B_2 \\
0 & 0 & A_3 - \lambda_j^{*1} I & E_{33} & \cdots & E_{3,\gamma-1} & B_3 \\
0 & 0 & 0 & A_4 - \lambda_j^{*1} I & \cdots & E_{4,\gamma-1} & B_4 \\
\vdots & \vdots & \vdots & \vdots & & \vdots & \vdots \\
0 & 0 & 0 & 0 & & E_{\gamma-1,\gamma-1} & B_{\gamma-1} \\
0 & 0 & 0 & 0 & & A_\gamma - \lambda_j^{*1} I & B_\gamma
\end{bmatrix} = \sum_{i=1}^{\gamma} n_i \;, \quad j=1,2,\ldots,n_1^*$$

and the system (12) is detectable if and only if the following conditions all hold:

$$(C1)' \qquad \operatorname{rank}\begin{bmatrix} A_1-\overset{*}{\lambda}{}^{1}_{j}I \\ C_1 \end{bmatrix} = n_1 \ , \ j=1,2,\ldots,\overset{*}{n}_1$$

$$(C2)' \qquad \operatorname{rank}\begin{bmatrix} A_1-\overset{*}{\lambda}{}^{2}_{j}I & E_{11} \\ 0 & A_2-\overset{*}{\lambda}{}^{2}_{j}I \\ C_1 & C_2 \end{bmatrix} = n_1+n_2, \ j=1,2,\ldots,\overset{*}{n}_2$$

$$(C3)' \qquad \operatorname{rank}\begin{bmatrix} A_1-\overset{*}{\lambda}{}^{3}_{j}I & E_{11} & E_{12} \\ 0 & A_2-\overset{*}{\lambda}{}^{3}_{j}I & E_{22} \\ 0 & 0 & A_3-\overset{*}{\lambda}{}^{3}_{j}I \\ C_1 & C_2 & C_3 \end{bmatrix} = n_1+n_2+n_3 \ , \ j=1,2,\ldots,\overset{*}{n}_3 \qquad (14)$$

$$\vdots \qquad\qquad \vdots \qquad\qquad \vdots$$

$$(C\gamma)' \quad \operatorname{rank}\begin{bmatrix} A_1-\overset{*}{\lambda}{}^{\gamma}_{j}I & E_{11} & E_{12} & E_{13} & \cdots & E_{1,\gamma-1} \\ 0 & A_2-\overset{*}{\lambda}{}^{\gamma}_{j}I & E_{22} & E_{23} & \cdots & E_{2,\gamma-1} \\ 0 & 0 & A_3-\overset{*}{\lambda}{}^{\gamma}_{j}I & E_{33} & \cdots & E_{3,\gamma-1} \\ 0 & 0 & 0 & A_4-\overset{*}{\lambda}{}^{\gamma}_{j}I & \cdots & E_{4,\gamma-1} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & E_{\gamma-1,\gamma-1} \\ 0 & 0 & 0 & 0 & \cdots & A_\gamma-\overset{*}{\lambda}{}^{\gamma}_{j}I \\ C_1 & C_2 & C_3 & C_4 & \cdots & C_\gamma \end{bmatrix} = \sum_{i=1}^{\gamma} n_i \ , \quad j=1,2,\ldots,\overset{*}{n}_\gamma$$

Similar results can be obtained for the corresponding feedback multi-composite system (as was done in Theorem 2).

The following Theorems shows that the series-parallel multi-composite system (12) is stabilizable and detectable for "almost all" [10] interconnection matrices, provided some mild conditions are imposed on the subsystems.

Theorem 6

Assume that (A_γ, B_γ) is stabilizable and that (C_1, A_1) is detectable and that A_i, $i = 1,2,\ldots,\gamma$ each have distinct eigenvalues; then:

(a) The class of all matrices $\{E_{ij},\ i=1,2,\ldots,\gamma-1\ ,\ j=i,i+1,\ldots,\gamma-1;\ C_2, C_3,\ldots,C_\gamma\ ;\ B_1,\ B_2,\ldots,B_{\gamma-1}\}$ which makes (12) not both stabilizable and detectable is either an empty set or a subset of a hypersurface in the parameter space of $\{E_{ij}, i=1,2,\ldots,\ldots,\gamma-1, j=i, i+1,\ldots,\gamma-1; C_2,\ldots,C_\gamma; B_1,\ldots,B_{\gamma-1}\}$.

(b) Let $B_1=B_2=\ldots=B_{\gamma-1}=0$; $C_2=C_3=\ldots=C_\gamma=0$; then the class of all matrices $\{E_{ij}, i=1,2,\ldots,\gamma-1, j=i, i+1,\ldots,\gamma-1\}$ which makes (12) not both stabilizable and detectable is either empty or a subset of a hypersurface in the parameter space of $\{E_{ij}, i=1,2,\ldots,\gamma-1, j=i, i+1,\ldots,\gamma-1\}$.

Consider now the completely general system:

$$\dot{x} = \begin{bmatrix} A_1 & E_{11} & E_{12} & E_{13} & \cdots & E_{1,\gamma-1} \\ F_{11} & A_2 & E_{22} & E_{23} & \cdots & E_{2,\gamma-1} \\ F_{12} & F_{22} & A_3 & E_{33} & \cdots & E_{3,\gamma-1} \\ F_{13} & F_{23} & F_{33} & A_4 & \cdots & E_{4,\gamma-1} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ F_{1,\gamma-1} & F_{2,\gamma-1} & F_{3,\gamma-1} & F_{4,\gamma-1} & \cdots & A_\gamma \end{bmatrix} x + \begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ B_4 \\ \vdots \\ B_\gamma \end{bmatrix} u \tag{15}$$

$$y = (C_1 \quad C_2 \quad C_3 \quad C_4 \quad \cdots \quad C_\gamma)\, x + D\, u$$

The following result is now obtained:

Theorem 7

Assume the same conditions as in Theorem 6; then:

(a) The class of all matrices $\{E_{ij}, F_{ij}, i=1,2,\ldots,\gamma-1, j=i,i+1,\ldots,\gamma-1; C_2, C_3,\ldots,C_\gamma; B_1, B_2,\ldots,B_{\gamma-1}\}$ which makes (15) not both stabilizable and detectable is either empty or a subset of a hypersurface in the parameter space of $\{E_{ij}, F_{ij}, i=1,2,\ldots,\gamma-1, j=i, i+1,\ldots,\gamma-1; C_2,\ldots,C_\gamma; B_1,\ldots,B_{\gamma-1}\}$.

(b) Let $B_1=B_2=\ldots=B_{\gamma-1}=0$, $C_2=C_3=\ldots=C_\gamma=0$; then the class of all matrices $\{E_{ij}, F_{ij}, i=1,2,\ldots,\gamma-1, j=i, i+1,\ldots,\gamma-1\}$ which makes (15) not both stabilizable and detectable is either empty or a subset of a hypersurface in the parameter space of $\{E_{ij}, F_{ij}, i=1,2,\ldots,\gamma-1, j=i, i+1,\ldots,\gamma-1\}$.

By using the same arguments as in the proof of all the previous Theorems, one can easily obtain the following more general result:

The following definitions are required:

Definition

Let S be any nonempty symmetric open subset of the complex plane C (i.e. if $\lambda \in S$ then its complex conjugate $\lambda^* \in S$).

Definition

Let all the eigenvalues of A_i, $i=1,2,\dots,\gamma$ not contained in S be denoted by ${}_s\lambda^i_j$, $j=1,2,\dots,{}_sn_i$, $i = 1,2,\dots,\gamma$. i.e.

$$\{{}_s\lambda^i_j \ , \ j=1,2,\dots,{}_sn_i, \ i=1,2,\dots,\gamma\} \notin S \qquad (16)$$

Corollary 1

Consider Theorems 1,2,5 ; then, the necessary and sufficient conditions for there to exist a time-invariant compensator for (1), (6) or (12) respectively such that the poles of the resultant closed loop system are all contained in S, are the same as the conditions previously stated in Theorems 1,2,5 with ${}_s\lambda^i_j$ replacing λ^{*i}_j and ${}_sn_i$ replacing n^*_i .

Remark 9

The above Corollary implies, in particular, that Theorem 5 is valid if "stabilizable", "detectable" is replaced by "controllable", "observable" respectively, provided that λ^{*i}_j ; n^*_i is replaced by λ^i_j , n_i respectively.

IV. CONCLUSIONS

Necessary and sufficient conditions have been obtained for a general series-parallel or feedback composite system, which includes hierarchial systems as a special case, to be either stabilizable or/and detectable. The conditions obtained are expressed in terms of the parameters of the state equations of the subsystems and in terms of the interconnecting parameters of the composite system, and are extremely simple to compute. Using these results it is shown that a composite system is stabilizable or/and detectable for "almost all" interconnection parameters provided some mild conditions on the system are imposed.

V. REFERENCES

[1] Rosenbrock H.H., State-space and Multivariable Theory, Nelson and Sons, 1970.

[2] Gilbert E.G., "Controllability and Observability in Multivariable Control Systems", SIAM J. Control, Vol. 1, 1963, pp. 128-151.

[3] Chen C.T., Desoer C.A., "Controllability and Observability of Composite Systems", IEEE Trans. on Automatic Control, Vol. AC-12, 1967, pp. 402-409.

[4] Bhandarkar M.V., Fahmy M.M., "Controllability of Tandem Connected Systems", IEEE Trans. on Automatic Control, Vol. AC-17, 1972, pp. 150-151.

[5] Wang S.H., Davison E.J., "On the Controllability and Observability of

Composite Systems", IEEE Trans. on Automatic Control, Vol. AC-18, 1973, p. 74-75.

[6] Wolovich W.A., Hwang H.L., "Composite System Controllability & Observability" Automatica, Vol.10, No.2, 1974, pp. 209-212.

[7] Hautus M.L.J., "Input Regularity of Cascaded Systems", IEEE Trans. on Automatic Control, to appear.

[8] Davison E.J., Wang S.H., "New Results on the Controllability & Observability of General Composite Systems", Department of Electrical Engineering, University of Toronto, Control System Report No. 7404 Jan. 1974, IEEE Trans. on Automatic Control, to appear.

[9] Davison E.J., "The Stabilizability of Large Interconnected Systems", Department of Electrical Engineering, University of Toronto, Control System Report No.7420 September 1974.

INTERCONNECTION OF UNSTABLE LINEAR SYSTEMS

C.A. DESOER AND W. S. CHAN
Department of Electrical Engineering and Computer
Sciences and the Electronics Research Laboratory
University of California, Berkeley, California 94720

ABSTRACT

In this preliminary report we consider the feedback interconnection of unstable, multivariable, linear, time-invariant systems. Since our methods are purely algebraic, the conclusions apply equally well to continuous-time as to discrete-time systems. In principle one has to test the stability of a 2n×2n matrix. We give three theorems where the tests require only the consideration of two n×n matrices. We give a complete description of the source of the instability for the n=1 case and give example of the 10 possible instability configurations. Finally four equivalent forms of the characteristic polynomial are given.

I. INTRODUCTION

In applications the vast majority of systems can be thought as an interconnection of subsystems. This is a result of the design procedure where engineers add various subsystems at various places to fulfill conflicting specifications, or is a result of the conceptualization of the overall system as in the case of economic and ecological systems. The main interest in our investigation is stability, more precisely input-output relations. We start with the basic feedback configuration, [1]. This preliminary report emphasizes the interconnection of unstable subsystems. It develops a number of new results for the case where all subsystems are linear, lumped and time-invariant. Therefore the techniques are purely algebraic and, hence, apply equally well to the continuous-time as to the discrete-time case.

We use standard notations. Capital symbols $\hat{G}_1$, $\hat{H}_e$ denote matrices; lower case symbols $\hat{g}_1$, $\hat{g}_2$ denote scalars. The superscript ^ denotes Laplace transforms.

II. SYSTEM DESCRIPTION

We consider a feedback system S_{LL} made up of two n-input n-output subsystems with transfer functions $\hat{G}_k(s) \in \mathbb{R}^{n\times n}(s)$, (i.e. elements are rational functions) which are proper (bounded at infinity), k = 1,2.

The inputs u_i, errors e_i, outputs y_i are functions from $\mathbb{R}_+$ into $\mathbb{R}^n$. We have

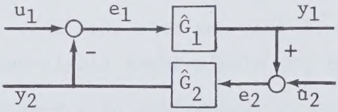


Fig. I.

$$\begin{bmatrix} \hat{e}_1(s) \\ \hat{e}_2(s) \end{bmatrix} = \begin{bmatrix} \hat{H}_e(s) \end{bmatrix} \begin{bmatrix} \hat{u}_1(s) \\ \hat{u}_2(s) \end{bmatrix} , \qquad (1)$$

$$\hat{H}_e(s)^{-1} = \begin{bmatrix} I & \hat{G}_2(s) \\ -\hat{G}_1(s) & I \end{bmatrix} \qquad (2)$$

Hence, after a few computations,

Research sponsored by the National Science Foundation Grant GK-10656X3.

$$\hat{H}_e(s) = \begin{bmatrix} (I+\hat{G}_2\ \hat{G}_1)^{-1} & -\hat{G}_2(I+\hat{G}_1\ \hat{G}_2)^{-1} \\ \hat{G}_1(I+\hat{G}_2\ \hat{G}_1)^{-1} & (I+\hat{G}_1\ \hat{G}_2)^{-1} \end{bmatrix} \tag{3}$$

We assume throughout that

$$\det[I+\hat{G}_2(\infty)\hat{G}_1(\infty)] = \det[I+\hat{G}_1(\infty)\hat{G}_2(\infty)] \neq 0 \tag{3a}$$

We say that a transfer function $\in \mathbb{R}^{n\times n}(s)$ is exponentially stable iff it is proper and has all its poles in $\mathring{\mathbb{C}}_-$, (open left half plane). By unstable we mean not exponentially stable.

By examples it can be shown that the four submatrices in $\hat{H}_e$ can display the $16=2^4$ possible patterns of stability and instability. In particular, any one of the $n\times n$ submatrices of $\hat{H}_e$ can be unstable while the other three are exp. st.. See Examples 8 and 9 in Appendix II.

III. SIMPLIFYING THEOREMS

These examples show that to discuss the exponential stability of the system S_{LL} of Fig. I, it is not enough to consider only the transfer function $\hat{G}_1(I+\hat{G}_2\ \hat{G}_1)^{-1}$: $u_1 \mapsto y_1$. In fact if $\hat{G}_2$ is exp. st., then the exp. st. of $\hat{G}_1(I+\hat{G}_2\ \hat{G}_1)^{-1}$ guarantees that of $\hat{H}_e$.

Theorem I. (Stable feedback subsystem). For the system S_{LL} defined above,

$$\hat{G}_2 \text{ and } \hat{G}_1(I+\hat{G}_2\ \hat{G}_1)^{-1} \text{ are exp. st.} \tag{4}$$

$$\Rightarrow \hat{H}_e \text{ is exp. st.} \tag{5}$$

When $\hat{G}_2$ is unstable one must in principle check the four submatrices in (3) for exp. st.. There are however some simplifications.

Theorem II. (SISO). For n=1 and the system S_{LL} described above, we have

$$\hat{g}_1(1+\hat{g}_2\ \hat{g}_1)^{-1} \text{ and } \hat{g}_2(1+\hat{g}_1\ \hat{g}_2)^{-1} \text{ are exp. st.} \tag{6}$$

$$\Leftrightarrow \hat{H}_e \text{ is exp. st.} \tag{7}$$

Theorem II does not hold when n>1.

For the single-input single-output case, (n=1), the origin of the stability or instability of a transfer function of the form $\hat{g}_1(1+\hat{g}_2\hat{g}_1)^{-1}$ can be completely described. To accomplish this we introduce some concepts.

A list is defined to be a set of order pairs $\in \mathbb{C} \times \mathbb{Z}_+$. For $\hat{g} \in \mathbb{R}(s)$, let $\mathcal{P}[\hat{g}]$, $(\mathcal{Z}[\hat{g}])$, denote the list where first element of each of its ordered pairs, say α, is the location of a pole, (zero, respectively), and the corresponding second element is the multiplicity of that pole, say m_α, (zero, respectively). By convention, $\forall\alpha \in \mathbb{C}$, we consider that $(\alpha,0) \in \mathcal{P}[\hat{g}]$, $(\mathcal{Z}[\hat{g}])$, because $(\alpha,0) \in \mathcal{P}[\hat{g}]$, $(\mathcal{Z}[\hat{g}])$, is equivalent to α is not a pole of $\hat{g}$, (a zero of $\hat{g}$, respectively). Two lists are defined to be equal if and only if they are equal as two sets of ordered pairs. We define list-inclusion relationship and binary operations on the lists as follows. Let A and B be lists.

$$A \textcircled{c} B \overset{\Delta}{\Leftrightarrow} \forall(\alpha,\ m_{\alpha A}) \in A, \exists(\alpha,\ m_{\alpha B}) \in B \ni$$

$$m_{\alpha A} \leq m_{\alpha B}$$

$$A \textcircled{$\cup$} B \triangleq \{(\alpha,\ m_{\alpha A} + m_{\alpha B}):\ (\alpha,\ m_{\alpha A}) \in A,\ (\alpha,\ m_{\alpha B}) \in B \text{ where either } m_{\alpha A} \text{ or } m_{\alpha B} \text{ may be zero}\}$$

$$A \textcircled{$\cap$} B \triangleq \{(\alpha,\ \min\{m_{\alpha A},\ m_{\alpha B}\}):\ (\alpha,\ m_{\alpha A}) \in A,\ (\alpha,\ m_{\alpha B}) \in B \text{ where either } m_{\alpha A} \text{ or } m_{\alpha B} \text{ may be zero}\}$$

$$A \ominus B \triangleq \{(\alpha,\ \max\{m_{\alpha A} - m_{\alpha B},\ 0\}):\ (\alpha,\ m_{\alpha A}) \in A,\ (\alpha,\ m_{\alpha B}) \in B \text{ where either } m_{\alpha A} \text{ or } m_{\alpha B} \text{ may be zero}\}$$

Theorem III. (SISO).

Let $\hat{g}_1,\ \hat{g}_2 \in \mathbb{R}(s)$ and be proper, then

$$\mathcal{P}[\hat{g}_1/(1 + \hat{g}_2\hat{g}_1)] = \left(\mathcal{Z}[1 + \hat{g}_2\hat{g}_1] \ominus \mathcal{Z}[\hat{g}_1]\right) \textcircled{$\cup$} \left(\mathcal{P}[\hat{g}_1] \textcircled{$\cap$} \mathcal{Z}[\hat{g}_2]\right)$$

For the multiple-input multiple-output, (MIMO), case there is another simplifying condition. It is given in the theorem that follows.

Theorem IV. (MIMO). For the system S_{LL} described above, if

$$\hat{G}_1 \text{ and } \hat{G}_2 \text{ have no common pole in } \mathbb{C}_+ \tag{12}$$

then

$$\text{the exponential stability of } \hat{G}_1(I + \hat{G}_2\hat{G}_1)^{-1} \text{ and } \hat{G}_2(I + \hat{G}_1\hat{G}_2)^{-1} \tag{13}$$

implies that

$$\hat{H}_e \text{ is exp. st.} \tag{14}$$

IV. THE CHARACTERISTIC POLYNOMIAL OF S_{LL}

Let $\mathbb{R}^{n\times n}[s]$ denote the noncommutative ring of polynomials in s with coefficients in $\mathbb{R}^{n\times n}$.

For k = 1,2, let

$$\hat{G}_k = D_{k\ell}^{-1} N_{k\ell} = N_{kr} D_{kr}^{-1} \tag{25}$$

be a left-coprime factorization and right-coprime factorization of $\hat{G}_k$. As a consequence [1], [2], [3], [4]:

$$N_{k\ell},\ N_{kr},\ D_{kr},\ D_{k\ell} \in \mathbb{R}^{n\times n}[s] \tag{26}$$

$$\operatorname{rank}\begin{bmatrix} D_{kr}(s) \\ N_{kr}(s) \end{bmatrix} = \operatorname{rank}\begin{bmatrix} N_{k\ell}(s) & D_{k\ell}(s) \end{bmatrix} = n \qquad \forall s \in \mathbb{C} \tag{27}$$

and (27) is a sufficient condition for the factorization in (25) to be coprime. Furthermore,

$$p \text{ is a pole of } \hat{G}_k \Leftrightarrow \det D_{k\ell}(p) = 0 \Leftrightarrow \det D_{kr}(p) = 0, \tag{28}$$

and

$$z \text{ is a zero of } \hat{G}_k \Leftrightarrow \det N_{k\ell}(z) = 0 \Leftrightarrow \det N_{kr}(z) = 0. \tag{29}$$

Recall that (27) expresses the fact that the factorization is right-coprime and left-coprime, resp.

Since for each $\hat{G}_k$ there are two factorizations, hence equ. (2) leads to four expressions for $\hat{H}_e$:

$$\hat{H}_e = \begin{bmatrix} I & \hat{G}_2 \\ -\hat{G}_1 & I \end{bmatrix}^{-1} \tag{30}$$

$$= \begin{bmatrix} I & N_{2r}D_{2r}^{-1} \\ -N_{1r}D_{1r}^{-1} & I \end{bmatrix}^{-1} = \begin{bmatrix} D_{1r} & 0 \\ 0 & D_{2r} \end{bmatrix} \begin{bmatrix} D_{1r} & N_{2r} \\ -N_{1r} & D_{2r} \end{bmatrix}^{-1} \tag{31}$$

$$= \begin{bmatrix} I & D_{2\ell}^{-1}N_{2\ell} \\ -D_{1\ell}^{-1}N_{1\ell} & I \end{bmatrix}^{-1} = \begin{bmatrix} D_{2\ell} & N_{2\ell} \\ -N_{1\ell} & D_{1\ell} \end{bmatrix}^{-1} \begin{bmatrix} D_{2\ell} & 0 \\ 0 & D_{1\ell} \end{bmatrix} \tag{32}$$

Using (27), we can easily show that the right-hand sides of (31) and (32) are right-coprime and left coprime, resp. By (28),

$$p \in \mathbb{C} \text{ is a pole of } \hat{H}_e \Leftrightarrow \det \begin{bmatrix} D_{1r} & N_{2r} \\ -N_{1r} & D_{2r} \end{bmatrix}(p) = 0 \tag{33}$$

$$\Leftrightarrow \det \begin{bmatrix} D_{2\ell} & N_{2\ell} \\ -N_{1\ell} & D_{1\ell} \end{bmatrix}(p) = 0 \tag{34}$$

We also have

$$\hat{H}_e = \left[\begin{array}{c|c} I & N_{2r}D_{2r}^{-1} \\ \hline -D_{1\ell}^{-1}N_{1\ell} & I \end{array}\right]^{-1} = \begin{bmatrix} I & 0 \\ 0 & D_{2r} \end{bmatrix} \left[\begin{array}{c|c} I & N_{2r} \\ \hline -N_{1\ell} & D_{1\ell}D_{2r} \end{array}\right]^{-1} \begin{bmatrix} I & 0 \\ 0 & D_{1\ell} \end{bmatrix} \tag{35}$$

$$= \left[\begin{array}{c|c} I & D_{2\ell}^{-1}N_{2\ell} \\ \hline -N_{1r}D_{1r}^{-1} & I \end{array}\right]^{-1} = \begin{bmatrix} D_{1r} & 0 \\ 0 & I \end{bmatrix} \left[\begin{array}{c|c} D_{2\ell}D_{1r} & N_{2\ell} \\ \hline -N_{1r} & I \end{array}\right]^{-1} \begin{bmatrix} D_{2\ell} & 0 \\ 0 & I \end{bmatrix} \tag{36}$$

Using (27), we can easily show that, in the right-hand side of (35) and (36), the first two factors are right coprime and the last two are left coprime and using a theorem of [5] and elementary row operations we see that

$$p \in \mathbb{C} \text{ is a pole of } \hat{H}_e \Leftrightarrow \det[D_{1\ell}D_{2r} + N_{1\ell}N_{2r}](p) = 0 \tag{37}$$

$$\Leftrightarrow \det[D_{2\ell}D_{1r} + N_{2\ell}N_{1r}](p) = 0. \tag{38}$$

Considering the four equivalences (33), (34), (37) and (38), and calling $\Omega \in \mathbb{R}[s]$ the monic polynomial whose zeros coincide with those of the polynomial in (33), (34), (37) and (38) we conclude with the

Theorem V. $p \in \mathbb{C}$ is a pole of $\hat{H}_e \Leftrightarrow \det \Omega(p) = 0$. [In other words, the polynomial Ω is the characteristic polynomial of the system S_{LL}. Therefore the algorithms that give the factorization [1], [2], [3] together with the Liénard-Chipart test, or graphical test, lead to techniques for testing the stability of S_{LL}.]

Important Remark: *The Discrete-Time Case*. Since all the techniques of this paper are purely algebraic in character, the results of this paper extend immediately to the discrete-time case. Then the u_i, e_i, y_i are sequences in $\mathbb{R}^n$, the Laplace transforms become z-transforms; $\mathbb{C}_+$ is replaced by $\overline{B}(0;1)$, the closed disk of center (0,0) and radius 1; the point at infinity of the s-plane becomes the point at infinity of the z-plane, e.g. in (3a).

Finally, the extension of these results to linear distributed systems and to nonlinear systems is in progress [6], [7].

REFERENCES

[1] C. A. Desoer and M. Vidyasagar, *Feedback Systems, Input-Output Properties*, Academic Press, New York, 1974 (In press).

[2] S. H. Wang, "Design of linear multivariable systems," Coll. Eng., Univ. California, Berkeley, Memo. ERL-M309, Oct. 1971.

[3] W. A. Wolovich, "The determination of state-space representations for linear multivariable systems," *Automatica*, *9*, pp. 97-106, 1973.

[4] H. H. Rosenbrock, *State-space and Multivariable Theory*, Wiley, New York, 1970.

[5] C. A. Desoer and J. D. Schulman, "Cancellations in multivariable continuous-time and discrete-time feedback systems treated by greatest common divisor extraction," *IEEE Trans. Automat. Contr.*, vol. AC-18, pp. 401-402, August 1973.

[6] F. M. Callier and C. A. Desoer, Work in progress.

[7] C. A. Desoer and W. S. Chan, Work in progress.

[8] C. A. Desoer and J. D. Schulman, "Zeros and poles of matrix transfer functions and their dynamical interpretation," *IEEE Trans. Circuits Syst.*, vol. CAS-21, pp. 3-8, January 1974.

APPENDIX I: PROOFS

Proof of Theorem I. To prove ⇒ note that (4) implies the exponential stability of

$$I - \hat{G}_2 \cdot \hat{G}_1(I + \hat{G}_2\hat{G}_1)^{-1} = (I + \hat{G}_2\hat{G}_1)^{-1}, \tag{A1}$$

and that of

$$I - \hat{G}_1(I + \hat{G}_2\hat{G}_1)^{-1} \cdot \hat{G}_2 = I - \hat{G}_1\hat{G}_2(I + \hat{G}_1\hat{G}_2)^{-1} = (I + \hat{G}_1\hat{G}_2)^{-1}; \tag{A2}$$

hence that of $\hat{G}_2 \cdot (I + \hat{G}_1\hat{G}_2)^{-1}$//.

Proof of Theorem II. ⇐ obvious. ⇒ by contradiction: suppose $(1+\hat{g}_1\hat{g}_2)^{-1}$ has a pole $p \in \mathbb{C}_+$. Since (6) holds, $\hat{g}_1(p) = \hat{g}_2(p) = 0$. Hence $1 + \hat{g}_1(p)\,\hat{g}_2(p) = 1^{(+)}$ which is a contradiction. Hence $(1+\hat{g}_1\hat{g}_2) = (1+\hat{g}_2\hat{g}_1)$ has no poles in $\mathbb{C}_+$, hence, in view of (3a), it is exp. st.//

Remark. The conclusion (+) does not follow when $n>1$. For a definition of a zero in the case $n>1$, see [4], [8]. For example $I+\hat{G}_1\hat{G}_2 = \mathrm{diag}[(s-1)/s, 2s/(s+1)]$.

Proof of Theorem III.

$$\mathcal{P}[\hat{g}_1/(1+\hat{g}_2\hat{g}_1)] = (\mathcal{Z}[1+\hat{g}_2\hat{g}_1] \ominus \mathcal{Z}[\hat{g}_1]) \textcircled{U} (\mathcal{P}[\hat{g}_1] \ominus \mathcal{P}[1+\hat{g}_2\hat{g}_1]) \tag{A4}$$

$$\begin{aligned}
&\mathcal{P}[\hat{g}_1] \ominus \mathcal{P}[1+\hat{g}_2\hat{g}_1] \\
&= \mathcal{P}[\hat{g}_1] \ominus \mathcal{P}[\hat{g}_2\hat{g}_1] \\
&= \mathcal{P}[\hat{g}_1] \ominus \{(\mathcal{P}[\hat{g}_1] \ominus \mathcal{Z}[\hat{g}_2]) \textcircled{$\cup$} (\mathcal{P}[\hat{g}_2] \ominus \mathcal{Z}[\hat{g}_1])\} \\
&= \mathcal{P}[\hat{g}_1] \ominus (\mathcal{P}[\hat{g}_1] \ominus \mathcal{Z}[\hat{g}_2]) \ominus (\mathcal{P}[\hat{g}_2] \ominus \mathcal{Z}[\hat{g}_1]) \\
&= (\mathcal{P}[\hat{g}_1] \textcircled{$\cap$} \mathcal{Z}[\hat{g}_2]) \ominus (\mathcal{P}[\hat{g}_2] \ominus \mathcal{Z}[\hat{g}_1]) \qquad \text{(A5)}
\end{aligned}$$

It is obvious that

$$\mathcal{P}[\hat{g}_1] \textcircled{$\cap$} \mathcal{Z}[\hat{g}_2] \;\; \textcircled{c} \;\; \mathcal{Z}[\hat{g}_2]$$

$$\mathcal{P}[\hat{g}_2] \ominus \mathcal{Z}[\hat{g}_1] \;\; \textcircled{c} \;\; \mathcal{P}[\hat{g}_2]$$

and

$$\mathcal{P}[\hat{g}_2] \textcircled{$\cap$} \mathcal{Z}[\hat{g}_2] = \emptyset$$

$\therefore$ from (A5) we have

$$\mathcal{P}[\hat{g}_1] \ominus \mathcal{P}[1+\hat{g}_2\hat{g}_1] = \mathcal{P}[\hat{g}_1] \textcircled{$\cap$} \mathcal{Z}[\hat{g}_2] \qquad \text{(A6)}$$

Substituting (A6) into (A4), we obtain the desired result.//

Proof of Theorem IV. We have

$I - \hat{G}_2 \cdot \hat{G}_1(I+\hat{G}_2\hat{G}_1)^{-1} = (I+\hat{G}_2\hat{G}_1)^{-1} = I - \hat{G}_2(I+\hat{G}_1\hat{G}_2) \cdot \hat{G}_1$ (12) and (13) imply that these three expressions have no $\mathbb{C}_+$ poles. Hence, with (3a), $(I+\hat{G}_2\hat{G}_1)^{-1}$ is exp. st.. If we interchange 1 and 2 in the above, we see that $(I+\hat{G}_1\hat{G}_2)^{-1}$ is exp. st.//

APPENDIX II: EXAMPLES

The closed loop transfer function $\hat{H}_e$, defined in (3), consists of four $n \times n$ submatrices. There are therefore $2^4=16$ possible patterns of exponential stability and instability. It turns out that all possible 16 patterns exist. However due to symmetry, these 16 patterns reduce to 10. Table I specifies the transfer functions $\hat{G}_1$ and $\hat{G}_2$ of the subsystems and Table II gives the resulting pattern of stability and instability.

TABLE I

EXAMPLE NO.	$\hat{G}_1$	$\hat{G}_2$
1	$1/(s+1)$	$-1/(s+1)$
2	$\begin{bmatrix} \frac{s+2}{s+1} & \frac{1}{s-1} \\ 0 & \frac{1}{s} \end{bmatrix}$	$\begin{bmatrix} \frac{1}{s-1} & \frac{1}{s} \\ 0 & \frac{s+2}{s+1} \end{bmatrix}$
3	$1/s$	$s/(s-1)$
4	$\begin{bmatrix} \frac{1}{s+1} & \frac{1}{s-2} \\ 0 & \frac{1}{s+2} \end{bmatrix}$	$\begin{bmatrix} \frac{1}{s+1} & 0 \\ 0 & \frac{s-2}{s+2} \end{bmatrix}$
5	$s/(s-1)$	$(s-1)/s$
6	$\begin{bmatrix} \frac{1}{s+1} & \frac{1}{s} \\ 0 & \frac{1}{s+1} \end{bmatrix}$	$\begin{bmatrix} \frac{1}{s} & \frac{1}{s+1} \\ 0 & \frac{1}{s+1} \end{bmatrix}$
7	$\begin{bmatrix} \frac{2}{s+1} & \frac{1}{s-1} \\ 0 & \frac{2}{s+1} \end{bmatrix}$	SAME AS $\hat{G}_1$
8	$1/s$	$s/(s+1)$
9	$\begin{bmatrix} \frac{s}{s+1} & \frac{1}{s} \\ 0 & \frac{1}{s} \end{bmatrix}$	$\begin{bmatrix} \frac{1}{s+1} & \frac{1}{s} \\ 0 & \frac{1}{s+1} \end{bmatrix}$
10	$2/s$	$(s+1)/(s-1)$

TABLE II

EXAMPLE NO. SUBMATRICES OF $\hat{H}_e$	1	2	3	4	5	6	7	8	9	10
$(I+\hat{G}_1\hat{G}_2)^{-1}$	U	S	U	S	S	S	U	S	S	S
$\hat{G}_2(I+\hat{G}_1\hat{G}_2)^{-1}$	U	U	S	S	U	U	S	S	S	S
$(I+\hat{G}_2\hat{G}_1)^{-1}$	U	U	U	U	S	U	U	S	U	S
$\hat{G}_1(I+\hat{G}_2\hat{G}_1)^{-1}$	U	U	U	U	U	S	S	U	S	S

S, (U), indicates that the corresponding submatrices is exponentially stable, (unstable, respectively).

ON THE METHOD OF VECTOR LYAPUNOV FUNCTIONS

V. LAKSHMIKANTHAM
University of Texas at Arlington, Arlington, Texas 76019

ABSTRACT

The method of vector Lyapunov functions consists of employing several Lyapunov-like functions and the theory of vectorial differential inequalities. This general comparison technic leads to a more flexible mechanism to study the qualitative properties of nonlinear differential systems. This method is also effective in discussing interconnected dynamical systems where a construction of a single Lyapunov function is difficult. However, an unpleasant fact in this approach is the requirement of quasimonotone property of the comparison system. An idea to get around this will be indicated which might be useful in applications. Also, an extension of this technic to abstract cones will be discussed.

1. INTRODUCTION

As is well known [6], employing a single Lyapunov function and the theory of scalar differential inequalities offers a useful mechanism to study a variety of qualitative problems of differential equations in a unified way. Nevertheless, when using this powerful technique for concrete problems, the main difficulty one faces is the lack of an algorithm for constructing a Lyapunov function for any given nonlinear system. It is therefore natural to ask whether it might be more advantageous in certain situations to use a vector Lyapunov function and the corresponding theory of vector differential inequalities. The answer is positive and this leads to a more flexible technique.

2. MAIN RESULTS

Consider the differential system

$$x' = f(t,x), \quad x(t_0) = x_0 \ , \tag{2.1}$$

where $f \in C[R^+ \times S_\rho, R^n]$. Let $V \in C[R^+ \times S_\rho, R_+^N]$. We define

$$D^+V(t,x) \equiv \limsup_{h \to 0^+} \frac{1}{h} \left[V(t+h, x+hf(t,x)) - V(t,x) \right] \tag{2.2}$$

for $(t,x) \in R^+ \times S_\rho$. The following result [6] plays an important role whenever we use vector Lyapunov functions.

Theorem 2.1.

Let $V \in C[R^+ \times S_\rho, R_+^N]$ and $V(t,x)$ be locally Lipschitzian in x. Assume that the vector function $D^+V(t,x)$ defined by (2.1) satisfies the inequality

$$D^+V(t,x) \le g(t,V(t,x)), \quad (t,x) \in R^+ \times S_\rho, \tag{2.3}$$

where $g \in C[R^+ \times R_+^N, R^N]$, and the function $g(t,u)$ is quasimonotone nondecreasing in u for each $t \in R^+$. Let $r(t,t_0,u_0)$ be the maximal solution of

$$u' = g(t,u), \quad u(t_0) = u_0 \ge 0,$$

existing on $[t_0,\infty)$. If $x(t,t_0,x_0)$ is any solution of (2.1) such that $V(t_0,x_0) \le u_0$, then, as far as $x(t,t_0,x_0)$ exists to the right of t_0,

$$V(t,x(t,t_0,x_0)) \le r(t,t_0,u_0).$$

Having this comparison theorem at our disposal, it is easy to discuss various results by the method of vector Lyapunov functions. See for example [3,5,6,8] and several references mentioned therein. Let us give a typical result that can be utilized as a tool in applications [6].

Theorem 2.2.

Suppose that

(i) $g \in C[R^+ \times R_+^N, R^N]$, $g(t,0) \equiv 0$ and $g(t,u)$ is quasimonotone nondecreasing in u for each $t \in R^+$;

(ii) $V \in C[R^+ \times S_\rho, R_+^N]$, $V(t,x)$ is locally Lipschitzian in x, and $\sum_{i=1}^{N} V_i(t,x) \to 0$ as $||x|| \to 0$ for each t;

(iii) $f \in C[R^+ \times S_\rho, R^n]$, $f(t,0) \equiv 0$ and for $(t,x) \in R^+ \times S_\rho$,

$$D^+V(t,x) \le g(t,V(t,x)).$$

Then the equistability of the trivial solution $u = 0$ of (2.3) implies the equistability of the trivial solution $x = 0$ of (2.1).

In Theorem 2.2, we have chosen

$$\sum_{i=1}^{N} V_i(t,x)$$

as a measure in computations. One could use [6,8] other suitable measures; for example, $\max_i V_i(t,x)$ or $Q(V_1(t,x), \ldots, V_N(t,x))$ where $Q: R_+^N \to R^+$ and $Q(u)$ is monotone nondecreasing in u.

The following example exhibits the fruitfulness of using vector Lyapunov functions.

Example. Consider the two systems

$$\begin{aligned} x' &= e^{-t}x + y \sin t - (x^3 + xy^2)\sin^2 t, \\ y' &= x \sin t + e^{-t}y - (x^2y + y^3)\sin^2 t. \end{aligned} \tag{2.4}$$

If we choose a single Lyapunov function given by $V = x^2 + y^2$, we arrive at

$$D^+V \leq 2(e^{-t} + |\sin t|)\, V.$$

Clearly the trivial solution of $u' = 2(e^{-t} + |\sin t|)u$ is not stable and hence we cannot deduce any information about the stability of the trivial solution of (2.4), although it is easy to check that u is stable. If, on the other hand, we choose two Lyapunov functions given by

$$V_1 = \frac{1}{2}(x+y)^2 \quad \text{and} \quad V_2 = \frac{1}{2}(x-y)^2,$$

we readily get

$$D^+V_1 \leq 2(e^{-t} + \sin t)V_1 \quad \text{and} \quad D^+u_2 \leq 2(e^{-t} - \sin t)V_2.$$

One easily checks that the assumptions of Theorem 2.2 are satisfied and therefore the trivial solution of (2.4) is stable.

Of course, the method of vector Lyapunov functions can be used in other situations more effectively. For example, the method has been used to consider other weaker types of stability and boundedness like conditional stability, mixed stability and also to give global results that can be applied in various ways [3,4,6,7,8].

Vector Lyapunov functions appear in a natural way in the study of multivariable systems [1,2] which are composed of relatively simple subsystems. The multivariability and other properties like multiconnection, composite structure and variety of the nature of subsystems, make the construction of a single Lyapunov function difficult for such interconnected systems. Consider the system of differential equations

$$x_i' = F_i(t,x_i), \quad x_i(t_0) = x_{i0}, \quad i = 1, \ldots, N, \tag{2.5}$$

where $F_i \in C[R^+ \times R^{ki}, R^{ki}]$. Let $G_i \in C[R^+ \times R^n, R^{ki}]$ where $n = \sum_{i=1}^{N} k_i$ and consider the interconnected system

$$x_i' = F_i(t,x_i) + G_i(t,x), \quad x_i(t_0) = x_{i0} \tag{2.6}$$

Here we have let $x = (x_{k_1}, \ldots, x_{k_N})$. If we define $||x|| = ||x_{k_1}|| + \ldots + ||x_{k_N}||$, we can write (2.6) in the form (2.1). On the basis of the aggregated stability properties of the subsystems (2.5), Lyapunov functions $V_i(t,x_i)$ may be constructed. Then making use of this constructed vector Lyapunov function and the nature of interactions between the subsystems, it is possible to predict the stability properties of the interconnected system (2.6) by the method of Vector Lyapunov functions.

An unpleasant fact in this approach is the requirement of quasimonotonicity of the comparison system $g(t,u)$. If $g(t,u) = Au$ where A is $N \times N$ matrix, this requirement implies that non-diagonal elements of A must be ≥ 0. Since quasimonotonicity of A is not a necessary condition for the matrix A to be a stability matrix, the limitation of this method is clear. An idea to get around this difficulty is as follows:

Suppose that $g(t,u)$ is not quasimonotone. Define

$$g_{i_0}(t,\, \rho_1, \, \ldots \, , \rho_{i-1} \, , u_i \, , \rho_{i+1} \, , \ldots \, , \rho_N) =$$

$$\max_{|\bar{u}_j - \bar{\bar{u}}_j| \leq \rho_j} |g_i(t,\bar{u}_1 \, , \ldots \, , \bar{u}_{i-1} \, ,u_i \, , \bar{u}_{i+1} \, , \ldots \, , \bar{u}_N)$$

$$- \; g_i(t,\, \bar{\bar{u}}_1 \, , \ldots \, , \bar{\bar{u}}_{i-1} \, , u_i \, , \bar{\bar{u}}_{i+1} \, , \ldots \, , \bar{\bar{u}}_N)|, \; i \neq j,$$

and

$$G_{i,\beta}(t,\, \rho_1, \, \ldots \, , \rho_{i-1} \, , u_i \, , \rho_{i+1} \, , \ldots \, , \rho_N) =$$

$$g_i(t,\, v_1 \, , \ldots \, , v_{i-1} \, , u_i \, , v_{i+1} \, , \ldots \, , v_N)$$

$$+ \, g_{i0}(t,\rho_1 - v_1 + \beta_1, \, \ldots \, , \rho_{i-1} - v_{i-1} + \beta_{i-1}, \, u_i, \, \rho_{i+1} - v_{i+1} + \beta_{i+1}, \, \ldots \, , \rho_N - v_N + \beta_N)$$

$$+ \, g_{i0}(t,\, \beta_1 \, , \ldots \, , \beta_{i-1}, \, u_i, \beta_{i+1} \, , \ldots \, , \beta_N),$$

for $\rho_i \geq v_i - \beta_i$, $v_i = v_i(t)$ given functions and $\beta_i > 0$, $i=1,2,\ldots,N$. It can be shown that

$$G_{i,\beta} \; t,\rho_1, \, \ldots \, , \rho_{i-1}, \, u_i, \, \rho_{i+1}, \, \ldots \, , \rho_N)$$

$$\geq \; g_i(t,\rho_1, \, \ldots \, , \rho_{i-1}, \, u_i, \, \rho_{i+1}, \, \ldots \, , \rho_N)$$

if $\rho_i \geq v_i - \beta_i$. The function G_β is now quasimonotone, by construction, and consequently, a convenient choice of $v_i = v_i(t)$ would yield useful information. That is to say that whenever

$$D\_m(t) \leq g(t,m(t)), \quad t \geq t_0,$$

holds, where $g(t,u)$ is not quasimonotone, if we know that $m(t) \geq v(t)$, then $m(t) \leq R(t,v)$, $t \geq t_0$, where $R_\beta(t,v)$ is the maximal solution of

$$u' = G_\beta(t,u), \quad u(t_0) = m(t_0),$$

and $R(t,v) = \lim_{\beta\to 0} R_\beta(t,v)$. This means that this idea depends heavily on the knowledge of a lower estimate of $m(t)$. Of course, one could always take $v(t) \equiv 0$, (since $m(t) \geq 0$) in which case we do not obtain the best estimate. The knowledge of a better lower bound of $m(t)$ yields a better estimate of $m(t)$. The success of this technique remains to be tested.

It appears that the difficulty we face concerning the quasimonotonicity requirement is due to the choice of the cone relative to the comparison system, namely the cone of nonnegative elements of R^N i.e., R^N_+. A possible answer to this problem appears to be choosing a convenient cone other than R^N_+ to work with in a given situation and this idea remains to be investigated. Recently [9], a comparison technique in abstract cones is developed and attempts are being made to extend Lyapunov's theory in abstract cones.

For a number of applications of the method of Vector Lyapunov method, see [1,2,3,5,6,7,8].

REFERENCES

1. Bailey, F.N., The application of Lyapunov's second method to interconnected systems. J. SIAM Control, 3 (1966), 443-462.
2. Grujic, L.T. and Siljak, D.D., Asymptotic stability and instability of large scale systems. IEEE Trans. Automatic Control (1973).
3. Habets, P. and Peiffer, K., Classification of stability like concepts and their study using vector Lyapunov functions. J.M.A.A., 44 (1973), 537-570.
4. Ladde, G.S. and Leela, S., Analysis of invariant sets. Annali di Metem. Pura ed Appl. XCIV (1972), 283-289.
5. Lakshmikantham, V., Several Lyapunov functions. Proce. Int. Conf. on Nonlinear oscillations, Kiev, USSR (1970).
6. Lakshmikantham, V. and Leela, S., Differential and integral inequalities, Vol. I (1969), Academic Press, New York.
7. Lakshmikantham, V. and Leela, S., Global results and stability of motion. Proce. Camb. Phi. Soc. 70 (1971), 95-102.
8. Matrosov, V.M., Vector Lyapunov functions in the analysis of nonlinear interconnected systems. Sympos. Math. Roma, 6, Mecaniea nonlineare stabilita 1970, (1971), 209-262.

9. Lakshmikantham, V., Mitchell, A.R. and Mitchell, R.W., Maximal and minimal solutions and comparison results for differential equations in abstract cones, Univ. of Texas, Arlington, Technical Report No. 13.

STABILITY ANALYSIS OF STOCHASTIC COMPOSITE SYSTEMS

ANTHONY N. MICHEL ROBERT D. RASMUSSEN
Department of Electrical Engineering and Engineering Research Institute
Iowa State University
Ames, Iowa 50010

ABSTRACT

New stability results for several classes of large-scale stochastic systems (also called composite or interconnected systems) are established. In all cases the objective is the same: to analyze large-scale systems in terms of their lower-order subsystems and in terms of their interconnecting structure. In order to demonstrate the usefulness of the present method, several specific examples are considered.

I. INTRODUCTION

The stability analysis of large-scale deterministic systems (also called composite or interconnected systems) has been considered in several reports (see, e.g., [1]-[11]). In a recent paper [12], stability results for a class of continuous-time parameter stochastic composite systems (described by Ito differential equations) and for a class of discrete-time parameter stochastic composite systems were established. For all systems considered in [12] it is assumed that the stochastic disturbances occur at the subsystems but not at the interconnecting structure of the composite systems. Furthermore, it is assumed in [12] that all subsystems, when isolated (i.e., disconnected) are either asymptotically stable in the large with probability one or exponentially stable in the large with probability one.

In the present paper the following extensions to the results of [12] are presented: (a) in addition to Wiener processes, system disturbances described by Poisson step processes and finite state Markov chains are considered; (b) stochastic disturbances may not only enter into the subsystem structure but also into the interconnecting structure of composite systems; (c) composite systems with stable as well as unstable subsystems are considered; and, (d) the present results yield not only conditions for stability with probability one but also conditions for stability in the quadratic mean, and stability in probability. In all cases the objective is the same: to analyze large-scale systems in terms of their lower-order subsystems and in terms of their interconnecting structure.

In order to demonstrate the usefulness of the method presented, a specific example is considered.

Although discrete parameter stochastic composite systems are not considered in this paper, the present method can readily be extended to such systems.

II. NOTATION

Let R^n denote the Euclidean n-space, let $|\cdot|$ denote the Euclidean norm, let $\|\cdot\|$ denote an arbitrary norm and let $x' = (x_1, ----, x_n)$ denote the transpose of $x \epsilon R^n$. Let $\lambda(A)$ denote an eigenvalue of a square matrix A and let the norm of a rectangular matrix D, induced by the Euclidean norm be denoted by $\|D\| = \sqrt{\lambda_{max}(D'D)}$, where D' is the transpose of D. Also, for the rectangular matrix $D = ((d_{ij}))$, let $\|D\|_m = (\sum_{i,j} d_{ij}^2)^{\frac{1}{2}}$.

Let $T = [0, \infty)$ and let $\{x_t, t \epsilon T\}$ denote a continuous-time parameter stocastic process of dimension n. Henceforth it is assumed that $x_o = x$ is known.

A real-valued function $\varphi(r)$ is said to belong to class K (i.e., $\varphi \epsilon K$) if it is defined, continuous, and strictly increasing over $0 \le r < \infty$, and if it vanishes at $r = 0$.

Systems are considered which may be represented by equations of the form

$$dx = f(x, t)dt + \sigma(x, t)d\xi \tag{1}$$

where $x \epsilon R^n$, $t \epsilon T$, $f: R^n \times T \to R^n$, $\sigma: R^n \times T \to R^{n \cdot m}$, and where $\{\xi_t, t \epsilon T\}$ is an *independent increment Markov process*. In the present paper two specific types of processes for (1) are considered: (a) *normalized m-dimensional Wiener processes* with independent components, denoted by $\xi_t = z_t$, and, (b) *normalized m-dimensional Poisson step processes* with independent components, denoted by $\xi_t = q_t$. It is assumed that $f(\cdot)$ and $\sigma(\cdot)$ are sufficiently smooth (see [13]-[15]) to insure for every $x_o = x$ the existence and uniqueness of a solution of (1) with probability one. The solution of (1) is denoted by $\{x_t, t \epsilon T\}$ (if $\xi_t = z_t$, then $\{x_t, t \epsilon T\}$ is called an *Ito process*). It is also assumed that $\{x_t = 0, t \epsilon T\}$ is the only equilibrium of (1).

For various definitions of *stability with probability one* (stability w.p.1), *stability in probability* (stability i.p.), and *stability in quadratic mean* (stability q.m.) of the equilibrium of (1), refer to ([14]-[16]). Stability results for (1) involve the existence of Lyapunov type functions $V: R^n \times T \to R^1$ which are in the domain of the *weak infinitesimal operator* $\tilde{A}_{Q_m}$ for (1) (see [14]-[16]), where

$$Q_m = \{x \epsilon R^n,\ t \epsilon T : V(x, t) < m, m > 0 \text{ a constant}\}. \tag{2}$$

If $\xi_t = z_t$, then $\tilde{A}_{Q_m}$ is determined by

$$\tilde{A}_{Q_m} V = \mathcal{L} V = \frac{\partial V}{\partial t} + \sum_i f_i'(x, t) \frac{\partial V}{\partial x_i} + \frac{1}{2} \sum_{i,j} S_{ij}(x, t) \frac{\partial^2 V}{\partial x_i \partial x_j}$$

$$= \frac{\partial V}{\partial t} + f(x, t)' \nabla_x V(x, t) + \frac{1}{2} \sum_{i,j} S_{ij}(x, t) \nabla_{x_i x_j} V(x, t) \tag{3}$$

where $S_{ij}(x, t)$ are defined by the matrix $((S_{ij}(x, t))) = \sigma(x, t)\, \sigma(x, t)'$.

If $\xi_t = q_t$ is a Poisson step process with independent components having jump frequency equal to $\alpha_j \Delta t + 0(\Delta t)$ and zero mean jump distribution $P_j(\cdot)$ (α_j is a constant and $0(\Delta t)$ denotes an infinitesimal of higher order than Δt) then

$$\tilde{A}_{Q_m} V = \mathcal{D}\ V = \frac{\partial V}{\partial t} + f(x, t)' \nabla_x V(x, t) + \sum_j \int [V(x + \sigma_j(x, t)\eta, t) - V(x, t)] \alpha_j P_j(d\eta) \tag{4}$$

where $\sigma_i(x, t)$ denotes the $i^{\underline{th}}$ column vector of $\sigma(x, t) = (\sigma_1(x, t) ---, \sigma_m(x, t))$.

Also considered are systems described by equations of the form

$$dx = f(x, t, \xi)dt \tag{5}$$

where $x \epsilon R^n$, $t \epsilon T$, $\xi_t = y_t \epsilon Y$ is a finite state homogeneous Markov chain, $f: R^n \times T \times Y \to R^n$ and where $\{x_t = 0, t \epsilon T\}$ is assumed to be the only equilibrium of (5). The function $f(\cdot)$ is assumed to be sufficiently smooth to insure the existence and uniqueness of the solution $\{x_t, t \epsilon T\}$ (see [14]-[16]). For (5) one has (see [16])

$$\tilde{A}_{Q_m} V = \mathcal{A}\ V = \frac{\partial V}{\partial t} + \sum_i f_i(x, t, y^k) \frac{\partial V(x, t, y^k)}{\partial x_i} + \sum_j \alpha_{kj} [V(x, t, y^j) - V(x, t, y^k)] \tag{6}$$

The constant α_{ij} in (6) is defined by $p_{ij}(\Delta t) = \alpha_{ij}\Delta t + O(\Delta t)$, $i \neq j$, where $p_{ij}(\Delta t)$ denotes the probability of the change $y^i \rightarrow y^j$ during the time Δt.

III. COMPOSITE SYSTEMS CONSIDERED

Composite or interconnected systems are considered which may be described by equations of the form

$$dw_i = f_i(w_i, t)dt + g_i(w_1, ---, w_\ell, t)dt + \sum_{j=1}^{\ell} \sigma_{ij}(w_j, t)d\xi_j, \; i = 1, ----, \ell \tag{7}$$

where $w_i \epsilon R^{n_i}$, $f_i : R^{n_i} \times T \rightarrow R^{n_i}$, $\sigma_{ij} : R^{n_j} \times T \rightarrow R^{n_i \cdot m_j}$, $f_j(w_j, t) = 0$, $g_i(w_1, ---, w_\ell, t) = 0$ and $\sigma_{ij}(w_j, t) = 0$ for all $t \epsilon T$ if and only if $w_j = 0$, $\xi_i \epsilon R^{m_i}$, ξ_{i_t} is an independent increment Markov process, and $g_i : R^{n_1} \times --- \times R^{n_\ell} \times T \rightarrow R^{n_i}$. Letting $\sum_{j=1}^{\ell} n_j = n$, $\sum_{j=1}^{\ell} m_j = m$, $x' = (w_1', ---, w_\ell') \epsilon R^n$, $f'(x, t) = [f_1(w_1, t)', ---, f_\ell(w_\ell, t)']$, $g(x, t) = [g_1(w_1, ---, w_\ell, t)', ---, g_\ell(w_1, ---, w_\ell, t)'] \overset{\Delta}{=} [g_1(x, t)', --- g_\ell(x, t)']$, $\sigma(x, t) = ((\sigma_{ij}(w_j, t)))$ and $\xi' = (\xi_1', ---, \xi_\ell')$, one can represent (7) equivalently as

$$dx = f(x, t)dt + \sigma(x, t)d\xi + g(x, t)dt \overset{\Delta}{=} F(x, t)dt + \sigma(x, t)d\xi, \tag{8}$$

where $f : R^n \times T \rightarrow R^n$, $\sigma : R^n \rightarrow R^{nm}$, $g : R^n \times T \rightarrow R^n$ and $\xi \epsilon R^m$.

System (8) which is of the same form as (1) is called a composite system $\mathcal{S}$ (or an interconnected system $\mathcal{S}$) having decomposition (7). It may be viewed as a nonlinear and time-varying interconnection of ℓ isolated subsystems $\mathcal{S}_i$ of the form

$$dw_i = f_i(w_i, t)dt + \sigma_{ii}(w_i, t)d\xi_i. \tag{9}$$

The unique solutions of (9) are denoted by $\{w_{i_t}, t \epsilon T\}$ with $w_{i_o} \overset{\Delta}{=} w_i$.

Also considered are composite systems of the form

$$dw_i = f_i(w_i, t, \xi_i)dt + g_i(w_1, ---, w_\ell, t, \xi_1, ---, \xi_\ell)dt, i = 1, ---, \ell \tag{10}$$

where $w_i \epsilon R^{n_i}$, $\xi_i \epsilon Y_i \subset R^{m_i}$, ξ_{i_t} is a finite state homogeneous Markov chain, $f_i : R^{n_i} \times T \times Y_i \rightarrow R^{n_i}$ and $g_i : R^{n_1} \times --- \times R^{n_\ell} \times T \times Y_1 \times -- \times Y_\ell \rightarrow R^{n_i}$. Defining f, x, g and ξ similarly as in (8), and letting $Y = Y_1 \times --- \times Y_\ell$, one can represent (10) equivalently as

$$dx = f(x, t, \xi)dt + g(x, t, \xi)dt \overset{\Delta}{=} F(x, t, \xi)dt \tag{11}$$

Equation (11) is clearly of the same form as (5). The isolated subsystems of composite system (11) are represented by

$$dw_i = f_i(w_i, t, \xi_i)dt, \quad i = 1, ---, \ell. \tag{12}$$

IV. MAIN RESULTS

For (9) real-valued functions $V_i(\cdot)$ are employed with domain of definition in $R^{n_i} \times T$ and for (12) real-valued functions $V_i(\cdot)$ are used with domain of definition in $R^{n_i} \times T \times Y_i$. Henceforth (9) and (12) are referred to as isolated subsystems $\mathcal{S}_i$ and (8) and (11) are called composite or interconnected systems $\mathcal{S}$.

Definition 1. Isolated subsystem $\mathcal{S}_i$ possesses Property A if there exists a function $V_i(\cdot)$, two radially unbounded functions $\psi_{i1}, \psi_{i2} \in K$ and a function $\psi_{i3} \in K$ such that $\psi_{i1}(|w_i|) \le V_i(\cdot) \le \psi_{i2}(|w_i|)$ and $\tilde{A}_{Q_{mi}} V_i(\cdot) \le -\psi_{i3}(|w_i|)$ for all $(w_i, t) \in Q_{mi}$, resp. $(w_i, t, y_i) \in Q_{mi}$, and $m_i > 0$ arbitrarily large.

Definition 2. Isolated subsystem $\mathcal{S}_i$ is said to possess Property B if there exists a function $V_i(\cdot)$, two radially unbounded functions $\psi_{i1}, \psi_{i2} \in K$ and two functions $\psi_{i3}, \psi_{i4} \in K$ such that $\psi_{i1}(|w_i|) \le V_i(\cdot) \le \psi_{i2}(|w_i|)$ and $\psi_{i4}(|w_i|) \le \tilde{A}_{Q_{mi}} V_i(\cdot) \le \psi_{i3}(|w_i|)$ for all $(w_i, t) \in Q_{mi}$, resp. $(w_i, t, y_i) \in Q_{mi}$, and $m_i > 0$ arbitrarily large.

Remark 1. These definitions imply, respectively, asymptotic stability and instability in an appropriate sense of isolated subsystems $\mathcal{S}_i$, described by (9) or (12) (see [14]-[17]).

The proofs of the subsequent results are given in Section VI.

Theorem 1. Assume that for composite system $\mathcal{S}$ (described by (7) or (10)), the following conditions hold:

(i) in (7), $\{\xi_{i_t} = z_{i_t}, t \in T\}$ or $\{\xi_{i_t} = q_{i_t}, t \in T\}$, and $\sigma_{ij}(w_j, t) = 0$ for all $i \ne j$; (ii) let $M = \{1, ---, \ell\}$ and let $N \subset M$. Each isolated subsystem $\mathcal{S}_i$ (described by (9) or (12), resp.) possesses Property A if $i \notin N$, $i \in M$, and Property B if $i \in N$; (iii) for each scalar product $\nabla_{w_i} V_i(\cdot)' g_i(\cdot)$, $i = 1, ---, \ell$ there exist continuous real-valued functions $a_{ij}(x, t)$, resp. $a_{ij}(x, t, y)$, such that,

$$\nabla_{w_i} V_i(\cdot)' g_i(\cdot) \le [\psi_{i3}(|w_i|)]^{\frac{1}{2}} \sum_{j=1}^{\ell} a_{ij}(\cdot) [\psi_{j3}(|w_j|)]^{\frac{1}{2}}$$

holds for all $w_i \in R^{n_i}$, $w_j \in R^{n_j}$, $x \in R^n$, $t \in T$, resp., $y_i \in Y_i$; (iv) there exists an ℓ-vector $\alpha' = (\alpha_1, ---, \alpha_\ell)$, $\alpha_i > 0$, $i = 1, ---, \ell$, and $\epsilon > 0$ such that for each $x \in R^n$ and $t \in T$, resp. $y \in Y$, the matrix $S + \epsilon I$ is negative definite, where I is the identity matrix and where $S = ((s_{ij}))$ is defined by $s_{ii} = \alpha_i [a_{ii}(\cdot) + 1]$, $i \in N$, $s_{ii} = \alpha_i [a_{ii}(\cdot) - 1]$, $i \notin N$ and $s_{ij} = [\alpha_i a_{ij}(\cdot) + \alpha_j a_{ji}(\cdot)]/2$, $i \ne j$. Then the equilibrium of composite system $\mathcal{S}$ (described by (7) or (10) is ASL w.p.1 and ASL i.p.

Corollary 1. Assume that for composite system $\mathcal{S}$ (described by (7) or (10) the following conditions hold:

(i) hypothesis (i) of Theorem 1 holds; (ii) the interconnecting structure of $\mathcal{S}$ is specified by $g_i(\cdot) = \sum_{j=1}^{\ell} g_{ij}(\cdot) w_j$; (iii) let $M = \{1, ---, \ell\}$ and let $N \subset M$. Each isolated subsystem $\mathcal{S}_i$ (described by (9) or (12) possesses Property B if $i \in N$ and Property A if $i \notin N$, $i \in M$ with $\psi_{ij}(|w_j|) = c_{ij} |w_j|^2 \ge 0$; (iv) for each $i, j = 1, ---, \ell$, there is a constant b_{ij} such that $\nabla_{w_i} V_i(\cdot)' g_{ij}(\cdot) \le b_{ij} |w_i| |w_j|$ for all $w_i \in R^{n_i}$, $w_j \in R^{n_j}$, $t \in T$, resp., $y_i \in Y_i$; (v) there exists an ℓ-vector $\alpha' = (\alpha_1, ---, \alpha_\ell)$, $\alpha_i > 0$, $i = 1, ---, \ell$, such that the matrix $S = ((s_{ij}))$ defined by $s_{ii} = \alpha_i (b_{ii} + c_{i3})$, $i \in N$, $s_{ii} = \alpha_i (b_{ii} - c_{i3})$, $i \notin N$, and $s_{ij} = [\alpha_i b_{ij} + \alpha_j b_{ji}]/2$, $i \ne j$, is negative definite. Then the equilibrium of composite system $\mathcal{S}$ (described by (7) or (10)) is ESL w.p.1, ESL q.m., and ESL i.p.

Theorem 2. Assume that for composite system (8) with decomposition (7) the following conditions hold:

(i) $\{\xi_{i_t} = z_{i_t}, t \epsilon T\}$ (i.e., ξ_{i_t} is a normalized Wiener process with independent components) and assume that in general $\sigma_{ij}(w_j, t) \neq 0$, $i, j = 1,---,\ell$; (ii) let $M = \{1,---,\ell\}$ and let $N \subset M$. Each isolated system $\mathcal{S}_i$ described by (9) possesses Property B if $i \epsilon N$ and Property A if $i \notin N$; (iii) for each scalar product $\nabla_{w_i} V_i(w_i, t)' g_i(x, t)$, $i = 1,---,\ell$, there exist continuous real-valued functions $a_{ij}(x, t)$ such that $\nabla_{w_i} V_i(w_i, t)' g_i(x, t) \leq [\psi_{i3}(|w_i|)]^{\frac{1}{2}} \sum_{j=1}^{\ell} a_{ij}(x, t)[\psi_{j3}(|w_j|)]^{\frac{1}{2}}$ hold for all $w_i \epsilon R^{n_i}$, $w_j \epsilon R^{n_j}$, $x \epsilon R^n$, and $t \epsilon T$; (iv) for each $V_i(w_i, t)$, $i = 1,---,\ell$, there is a positive constant e_i such that $u_i'[\nabla_{w_i w_i} V_i(w_i, t)]u_i \leq e_i |u_i|^2$ holds uniformly in w_i and $t \epsilon T$ for all $u_i \epsilon R^{n_i}$; (v) for each $i, j = 1,---,\ell$, $i \neq j$, there exists a continuous real-valued function $d_{ij}(w_j)$ such that $d_{ij}(w_j) \geq 0$ for all $w_j \epsilon R^{n_j}$, and such that $\|\sigma_{ij}(w_j, t)\|_m^2 \leq d_{ij}(w_j)\psi_{j3}(|w_j|)$ for all $w_j \epsilon R^{n_j}$ and $t \epsilon T$; and, (vi) there exists an ℓ-vector $\alpha' = (\alpha_1,---,\alpha_\ell)$, $\alpha_i > 0$, $i = 1,---,\ell$, and $\epsilon > 0$ such that for each $x \epsilon R^n$, $t \epsilon T$, the matrix $S + \epsilon I$ is negative definite, where $S = ((s_{ij}))$ is defined by $s_{ii}(x, t) = \alpha_i[a_{ii}(x,t) + 1] + \frac{1}{2}\sum_{\substack{k=1 \\ i\neq k}}^{\ell} \alpha_k e_k d_{ki}(w_i)$, $i \epsilon N$, $s_{ii}(x, t) = \alpha_i[a_{ii}(x, t) - 1] + \frac{1}{2}\sum_{\substack{k=1 \\ i\neq k}}^{\ell} \alpha_k e_k d_{ki}(w_i)$, $i \notin N$, and $s_{ij}(x, t) = [\alpha_i a_{ij}(x, t) + \alpha_j a_{ji}(x, t)]/2$, $i \neq j$. Then the equilibrium of composite system (8) is ASL w.p.1 and ASL i.p.

Corollary 2. Assume that for composite system (8) with decomposition (7) the following conditions hold:

(i) hypotheses (i) and (iv) of Theorem 2 hold; (ii) hypothesis (v) of Theorem 2 holds with $d_{ij}(w_j) \equiv d_{ij} \geq 0$; (iii) hypotheses (ii), (iii), and (iv) of Corollary 1 hold; (iv) there exists an ℓ-vector $\alpha' = (\alpha_1,---,\alpha_\ell)$, $\alpha_i > 0$, $i = 1,---,\ell$, such that the matrix $S = ((s_{ij}))$ defined by $s_{ii} = \alpha_i(b_{ii} + c_{i3}) + \frac{1}{2}\sum_{\substack{k=1 \\ i\neq k}}^{\ell} \alpha_k e_k d_{ki}$, $i \epsilon N$, $s_{ii} = \alpha_i(b_{ii} - c_{i3}) + \frac{1}{2}\sum_{\substack{k=1 \\ i\neq k}}^{\ell} \alpha_k e_k d_{ki}$, $i \notin N$, and $s_{ij} = [\alpha_i b_{ij} + \alpha_j b_{ji}]/2$, $i \neq j$, is negative definite. Then the equilibrium of composite system (8) is ESL w.p.1, ESL q.m. and ESL i.p.

Theorem 3. Assume that for composite system $\mathcal{S}$ (described by (7) or (10) the following conditions hold:

(i) hypotheses (i) and (ii) of Theorem 1 are staisfied; (ii) for each scalar product $\nabla_{w_i} V_i(\cdot)' g_i(\cdot)$, $i = 1,---,\ell$, there exist real constants a_{ij} such that (a) $a_{ij} \geq 0$ for all $i \neq j$, and, (b) $\nabla_{w_i} V_i(\cdot)' g_i(\cdot) \leq \sum_{j=1}^{\ell} a_{ij}\psi_{j3}(|w_j|)$ for all $x \epsilon R^n$, $w_i \epsilon R^{n_i}$, $w_j \epsilon R^{n_j}$, $t \epsilon T$, resp., $y_i \epsilon Y_i$; (iii) the matrix $S = ((s_{ij}))$ defined by $s_{ii} = -a_{ii} - 1$, $i \epsilon N$, $s_{ii} = -a_{ii} + 1$, $i \notin N$, $s_{ij} = -a_{ij}$ if $i \neq j$, has positive successive principal minors. Then the equilibrium of composite system $\mathcal{S}$ (described by (7) or (10)) is ASL w.p.1 and ASL i.p.

Theorem 4. Assume that for composite system (8) with decomposition (7) the following conditions hold:

(i) hypotheses (i), (ii), and (iv) of Theorem 2 are satisfied; (ii) for each scalar product $\nabla_{w_i} V_i(w_i, t)' g_i(x, t)$, $i = 1,---,\ell$, there exist real constants a_{ij} such that (a) $a_{ij} \geq 0$ for all $i \neq j$, and (b) $\nabla_{w_i} V_i(w_i, t)' g_i(x, t) \leq \sum_{j=1}^{\ell} a_{ij} \psi_{j3}(|w_j|)$ for all $x \epsilon R^n$, $w_i \epsilon R^{n_j}$, $t \epsilon T$; (iii) hypothesis (ii) of Corollary 2 holds; (iv) the matrix $S = ((s_{ij}))$ defined by $s_{ii} = -a_{ii} - 1$, $i \epsilon N$, $s_{ii} = -a_{ii} + 1$, $i \notin N$, and $s_{ij} = -a_{ij} - \frac{1}{2} e_i d_{ij}$ if $i \neq j$ has positive successive principal minors. Then the equilibrium of composite system (8) is ASL w.p.1 and ASL i.p.

V. EXAMPLE

Presently a specific example is considered

Example Consider the white noise driven system

$$\dot{x} = A(x, t)x + \sigma(x, t)v(t) \tag{13}$$

where $x \epsilon R^n$, $t \epsilon T$, $v(t)$ is a normalized vector "white noise" process of dimension n, $\sigma(x, t)$ is an $n \times n$ matrix of continuous scalar functions $\sigma_{ij}(x, t)$ and $A(x, t)$ is an $n \times n$ matrix of continuous scalar functions $a_{ij}(x, t)$. It is assumed that $\sigma_{ij}(x, t)$ is of the form

$$\sigma_{ij}(x, t) = \overline{\sigma}_{ij} x_j \tag{14}$$

where $\overline{\sigma}_{ij}$ is a constant, and that $a_{ii}(x, t) \leq -\delta$, $\delta > 0$, for all $i = 1, ---, n$, $x \epsilon R^n$ and $t \epsilon T$.

System (13) may be modeled by the Ito differential equation

$$dx_i = \sum_{j=1}^{m} [a_{ij}(x, t) + \frac{1}{2}\overline{\sigma}_{ij}\overline{\sigma}_{jj}] x_j dt + \sum_{j=1}^{n} \overline{\sigma}_{ij} x_j dz_j \qquad i = 1, ---, n \tag{15}$$

where $z' = (z_1, ---, z_m)$ and $\{z_t, t \epsilon T\}$ denotes a normalized n-dimensional Wiener process. The isolated subsystems of (15) are represented by

$$\mathcal{S}_i : dx_i = [a_{ii}(x, t) + \frac{1}{2}\overline{\sigma}_{ii}^2] x_i dt + \overline{\sigma}_{ii} x_i dz_i, \quad i = 1, ---, n \tag{16}$$

and the interconnecting structure of (15) is specified by

$$g_{ij}(x, t)dt + \overline{\sigma}_{ij} x_j dz_j \overset{\Delta}{=} \underset{i \neq j}{a_{ij}(x, t) x_j dt + \frac{1}{2}\overline{\sigma}_{ij}\overline{\sigma}_{jj} x_j dt + \overline{\sigma}_{ij} x_j dz_j}. \tag{17}$$

Now let $\overline{a}_{ij}(x, t) \overset{\Delta}{=} a_{ij}(x, t) + \frac{1}{2}\overline{\sigma}_{ij}\overline{\sigma}_{jj}$, let $b_{ii} = \sup_{R^n x T} \overline{a}_{ii}(x, t)$ for all $i = 1, ---, n$, let $b_{ij} = \sup_{R^n x T} |\overline{a}_{ij}(x, t)|$ for all $i \neq j$, $i, j = 1, ---, n$, and choose $V_i(x_i) = \lambda_i |x_i|$, where $\lambda_i > 0$ is a constant. Letting $\mathcal{L}_i$ denote the operation $\mathcal{L}$ with respect to (16), one has

$$\mathcal{L}_i V_i = \lambda_i \operatorname{sgn}[x_i] \overline{a}_{ii}(x, t) x_i = \lambda_i \overline{a}_{ii}(x, t) |x_i| \leq -\lambda_i |b_{ii}| |x_i| \tag{18}$$

for all $x \epsilon Q_m - (Q_\epsilon + \partial Q_\epsilon)$, $\epsilon > 0$ arbitrarily small (∂Q_ϵ denotes the boundary of the open set Q_ϵ). In other words, (18) holds everywhere except in an arbitrarily small neighborhood of the origin of R^n. Under these conditions, the theorems of Section IV can still be used to test stability (see p. 49 of [14]). Using the notation of Theorem 4, let $g_i(x, t) = \sum_{\substack{j=1 \\ i \neq j}}^{n} g_{ij}(x, t) = \sum_{\substack{j=1 \\ i \neq j}}^{n} [a_{ij}(x, t) x_j + \frac{1}{2}\overline{\sigma}_{ij}\overline{\sigma}_{jj} x_j]$. Then $\nabla_{x_i} V_i(x_i)' g_i(x, t) = \lambda_i \operatorname{sgn}(x_i) \sum_{\substack{j=1 \\ i \neq j}}^{n} \overline{a}_{ij}(x,t) x_j$ $\leq \sum_{\substack{j=1 \\ i \neq j}}^{n} \lambda_i |\overline{a}_{ij}(x, t)| |x_j| \leq \sum_{\substack{j=1 \\ i \neq j}}^{n} \lambda_i b_{ij} |x_j| = \sum_{\substack{j=1 \\ i \neq j}}^{n} |\lambda_i b_{ij} / \lambda_j b_{jj}| \psi_{j3}(|x_j|)$, where

$\psi_{j3}(|x_j|) = \lambda_j |b_{jj}| |x_j|$. Matrix S of Theorem 4 is now defined by $s_{ii} = 1$ and $s_{ij} = -|\lambda_i b_{ij}/\lambda_j b_{jj}|$ if $i \neq j$. This matrix has positive principal minors if there exist $\gamma_i > 0$ such that

$$1 - \sum_{\substack{i=1 \\ i \neq j}}^{n} |\lambda_i b_{ij}/\lambda_j b_{jj}| \geq |\gamma_j / b_{jj}| > 0, \quad j = 1, ---, n. \tag{19}$$

Recalling the definition of $\bar{a}_{ij}(x, t)$, one obtains from (19)

$$\inf_{R^n xT} |a_{jj}(x, t)| - \sum_{\substack{i=1 \\ i \neq j}}^{n} \frac{\lambda_i}{\lambda_j} \sup_{R^n xT} |a_{ij}(x, t)| - \frac{1}{2}(\bar{\sigma}_{jj}^{\;2} + \sum_{\substack{i=1 \\ i \neq j}}^{n} \frac{\lambda_i}{\lambda_j} |\bar{\sigma}_{ij}\bar{\sigma}_{jj}|) \geq \gamma_j > 0 \tag{20}$$

$$j = 1, ---, n.$$

In summary, if there exist λ_i, $i = 1, ---, n$, such that (20) holds, then the equilibrium of (13) is ASL w.p.1 and ASL i.p. (by Theorem 4).

If in particular $A(x, t) \equiv A = ((a_{ij}))$, for all $x \epsilon R^n$, $t \epsilon T$, where A is a constant matrix, then (20) assumes the form

$$|a_{jj}| - \sum_{\substack{i=1 \\ i \neq j}}^{n} \frac{\lambda_i}{\lambda_j} |a_{ij}| - \frac{1}{2}(\bar{\sigma}_{jj}^{\;2} + \sum_{\substack{i=1 \\ i \neq j}}^{n} \frac{\lambda_i}{\lambda_j} |\bar{\sigma}_{ij}\bar{\sigma}_{jj}|) > 0, \quad j = 1, ---, n \tag{21}$$

If (21) is satisfied with $\bar{\sigma}_{ij} = 0$ for all i and j, then the famous Hicks conditions encountered in the stability analysis of economic systems modeled by deterministic linear equations are satisfied [22], [23]. If (20) or (21) are satisfied, then corresponding Hicks stability conditions of economic systems modeled by equations of the form (15) are obtained.

Remark 2. The deterministic version of equation (13) or (15) and its qualitative (i.e., stability) analysis arises in a great variety of disciplines: transistor circuits (see e.g. [11], [24]), economics (see e.g. [25], [26]), biology (see, e.g., [27]). etc..

VI. PROOF OF MAIN RESULTS

Before giving the proofs of the present results some preliminary remarks are in order.

Preliminaries

1. In each proof Lyapunov functions of the form

$$V(\cdot) = \sum_{i=1}^{\ell} \alpha_i V_i(\cdot) \tag{22}$$

are employed, where $\alpha_i > 0$, $i = 1, ---, \ell$. Henceforth, let $\alpha' = (\alpha_1, ---, \alpha_\ell)$.

2. If in (7), $\{\xi_{i_t} = z_{i_t}, t \epsilon T\}$, then it is easily shown that

$$\tilde{A}V(x, t) = \mathcal{L}V(x, t) = \sum_{i=1}^{\ell} \alpha_i \mathcal{L}_i V_i(w_i, t) + \sum_{i=1}^{\ell} \alpha_i g_i(x, t)' \nabla_{w_i} V_i(w_i, t) +$$

$$+ \frac{1}{2} \sum_{\substack{i,j=1 \\ i \neq j}}^{\ell} \alpha_i \mathrm{tr}[\sigma_{ij}(w_j, t)' \nabla_{w_i w_i} V_i(w_i, t) \sigma_{ij}(w_j, t)] \tag{23}$$

where tr[A] denotes the trace of a matrix A, and where $\mathcal{L}_i$ denotes the operation $\mathcal{L}$ specified by (3) for the $i^{\underline{th}}$ isolated subsystem $\mathcal{S}_i$ characterized by (9).

3. Choose $V(\cdot)$ as in (22) and let $Q_{m_i} = \{w_i \epsilon R^{n_i}, t \epsilon T, \text{ resp. } y_i \epsilon Y_i : V(\cdot) < m_i\}$. Let $m_i = m/\alpha_i$, $m > 0$. Then $\alpha_i V_i(\cdot) \leq \sum_{j=1}^{\ell} \alpha_j V_j(\cdot) = V(\cdot) < m$, which implies that $(w_i, t, y_i) \epsilon Q_{m_i}$ whenever $(x, t, y) \epsilon Q_m$.

4. For composite system (7) with $\sigma_{ij}(w_j, t) \equiv 0$ for $i \neq j$, and for composite system (10) it is easily shown that for composite system $\mathcal{S}$ one has

$$\widetilde{A}_{Q_m}V(\cdot) = \sum_{i=1}^{\ell} \alpha_i \widetilde{A}_{Q_{m_i}} V_i(\cdot) + \sum_{i=1}^{\ell} \alpha_i g_i(\cdot)' \nabla_{w_i} V_i(\cdot) \tag{24}$$

where $\widetilde{A}_{Q_{m_i}} V_i(\cdot)$ denotes the infinitesimal operator for subsystem $\mathcal{S}_i$.

5. In the proofs of Theorems 3 and 4 the following result is used.

Theorem A (see, e.g., [20], [21]). If $A = ((a_{ij}))$ is a real $n \times n$ matrix such that $a_{ij} \leq 0$, $i \neq j$, and if all usccessive principal minors of A are positive, then A is nonsingular and all elements of A^{-1} are non-negative.

The proofs of the present results follow along similar lines. Therefore, only the proof of Theorem 4 is presented.

Proof of Theorem 4. Choose $V(\cdot)$ as in (22) and define Q_{mi} as above. Then $(x, t) \epsilon Q_m$ implies that $(w_i, t) \epsilon Q_{mi}$, $i = 1, ---, \ell$. Since each subsystem $\mathcal{S}_i$ possesses Property A or Property B, it follows that

$$\psi_1(|x|) \overset{\Delta}{=} \sum_{i=1}^{\ell} \alpha_i \psi_{i1}(|w_i|) \leq V(\cdot) \leq \sum_{i=1}^{\ell} \alpha_i \psi_{i2}(|w_i|) \overset{\Delta}{=} \psi_2(|x|). \tag{25}$$

for all $(x, t) \epsilon Q_m$ and $m > 0$ arbitrarily large, and thus, $V(\cdot)$ is positive definite, decrescent, and radially unbounded.

In view of hypotheses (i)-(iii) and in view of (23) one has

$$\begin{aligned} V(x, t) &= \sum_{i=1}^{\ell} \alpha_i \mathcal{L}_i V_i(w_i, t) + \sum_{i=1}^{\ell} \alpha_i g_i(w_i t)' \nabla_{w_i} V_i(w_i, t) + \\ &+ \frac{1}{2} \sum_{\substack{i,j=1 \\ i \neq j}}^{\ell} \alpha_i \mathrm{tr}[\sigma_{ij}(w_j, t)' \nabla_{w_i w_i} V_i(w_i, t) \sigma_{ij}(w_j, t)] \leq \sum_{\substack{i=1 \\ i \notin N}}^{\ell} -\alpha_i \psi_{i3}(|w_i|) + \\ &+ \sum_{\substack{i=1 \\ i \epsilon N}}^{\ell} \alpha_i \psi_{i3}(|w_i|) + \sum_{i,j=1}^{\ell} \alpha_i a_{ij} \psi_{j3}(|w_j|) + \frac{1}{2} \sum_{\substack{i,j=1 \\ i \neq j}}^{\ell} \alpha_i e_i \|\sigma_{ij}(w_j, t)\|_m^2 \leq \\ &\leq \sum_{\substack{i=1 \\ i \notin N}}^{\ell} \alpha_i [a_{ii} - 1] \psi_{i3}(|w_i|) + \sum_{\substack{i=1 \\ i \epsilon N}}^{\ell} \alpha_i [a_{ii} + 1] \psi_{i3}(|w_i|) + \\ &+ \sum_{\substack{i,j=1 \\ i \neq j}}^{\ell} \alpha_i [a_{ij} + \frac{1}{2} e_i d_{ij}] \psi_{j3}(|w_j|). \end{aligned}$$

Letting $u' = [\psi_{13}(|w_1|), ---, \psi_{\ell 3}(|w_\ell|)]$, and defining $S = ((s_{ij}))$ as in hypothesis (iv), one has $\widetilde{A}_{Q_m}V(\cdot) \leq -\alpha' Su$. Let $y' = \alpha' S$. Then $\widetilde{A}_{Q_m}V(\cdot) \leq -y'u$. By hypothesis (i), (ii) and (iii), $s_{ij} \leq 0$ for $i \neq j$ and S has positive successive principal minors for all $x \epsilon R^n$. Therefore, S^{-1} exists for all $x \epsilon R^n$ and all elements of S^{-1} are non-negative, by Theorem A. Thus, $\alpha = (S^{-1})'y$. Since each row and column of S^{-1} must contain at least one nonzero element one can always choose y, $y_i > 0$, $i = 1, ---, \ell$, so that $\alpha_i > 0$, $i = 1, ---, \ell$. Therefore $A_{Q_m}V(x, t) \leq -y'u < 0$ if $x \neq 0$, and $A_{Q_m}V(x, t) = 0$ if $x = 0$, and there exists a function $\psi_3 \epsilon K$ such that

$$A_{Q_m}V(x, t) \leq -\psi_3(|x|) \tag{26}$$

for all $(x, t) \epsilon Q_m$ and $m > 0$ arbitrarily large.

The conclusion of the theorem follows from (25) and (26).

VII. CONCLUDING REMARKS

At this point several comments are in order. 1) If $\sigma_{ij}(w_j, t) = 0$ for all $i,j = 1,---,\ell$, then Theorem 2 reduces to a deterministic result reported in [10] and Theorem 4 reduces to a deterministic result reported in [8]. In [12] the following special case of Theorem 2 is established: (a) $N = \phi$ (i.e., N is the empty set), (b) $\sigma_{ij}(w_j, t) \equiv 0$ for all $i \neq j$, and (c) only ASL w.p.1 is considered. 2) In [12] the following special case of Corollaries 1 and 2 is considered: (a) only composite system (8) is treated, (b) $N = \phi$ (i.e., N is empty), (c) $\sigma_{ij}(w_j, t) \equiv 0$ for all $i \neq j$. If $\sigma_{ij}(w_j, t) \equiv 0$ for all $i,j = 1,---,\ell$ and if only composite system (8) is considered, then Corollaries 1 and 2 reduce to corresponding deterministic results reported in [10]. Using the present approach, it is also possible to establish results for the stochastic version of the deterministic systems considered in [1], [3], and [6]. 3) Using the ideas presented herein, it is also possible to extend the results of [4], [7], and [10] to discrete parameter stochastic interconnected systems. This was partially accomplished in [12]. 4) In addition to the stochastic stability concepts considered presently, one can also apply the present method to stochastic composite systems using other types of stochastic stability concepts (see e.g., [18] and [19]).

REFERENCES

[1] F. N. Bailey, "The application of Lyapunov's second method to interconnected systems", *J. SIAM Contr.*, ser. A., vol. 3, pp. 443-462, 1966.

[2] A. A. Pointkovskii and L. D. Rutkovskaya, "Investigation of certain stability theory by the vector Lyapunov function method", *Automation and Remote Control*, vol. 10, pp. 1422-1429, 1967.

[3] W. E. Thompson, "Exponential stability of interconnected systems", *IEEE Trans. Automat. Contr.*, vol. 15, pp. 504-506, Aug. 1970.

[4] M. Araki, K. Ando, and B. Kondo, "Stability of sampled-data composite systems with many nonlinearities", *IEEE Trans. Automat. Contr.*, vol. 16, pp. 22-27, Feb. 1971.

[5] A. N. Michel and D. W. Porter, "Stability analysis of composite systems", *IEEE Trans. Automat. Contr.*, vol. 17, pp. 222-226, Apr. 1972.

[6] M. Araki and B. Kondo, "Stability and transient behavior of composite nonlinear systems", *IEEE Trans. Automat. Contr.*, vol. 17, pp. 537-541.

[7] Lj. T. Grujic and D. D. Siljak, "On stability of discrete composite systems", *IEEE Trans. Automat. Contr.*, vol. 18, pp. 522-524, Oct. 1973.

[8] Lj. T. Grujic and D. D. Siljak, "Asymptotic stability and instability of large-scale systems", *IEEE Trans. Automat. Contr.*, vol. 18, pp. 636-645, Dec. 1973.

[9] D. W. Porter and A. N. Michel, "Bounded input bounded output stability of nonlinear time-varying multiloop systems", *Pro. 1974 Joint Automat. Contr. Conf.*, Univ. of Texas, Austin, Tex., 1974, pp. 75-82. *IEEE Trans. Automat. Contr.*, vol. 19, Aug. 1974, pp. 422-427.

[10] A. N. Michel, "Stability analysis of interconnected systems", to appear.

[11] A. N. Michel, "Stability analysis and trajectory behavior of composite systems", Proc. 1974 IEEE Internat. Symp. on Circuits and Systems, San Francisco, Calif., 1974, pp. 240-244. To appear in IEEE Trans. on Circuits and Systems, vol. 22, April 1975.

[12] A. N. Michel, "Stability analysis of stochastic large-scale systems", Proc. Eighth Annual Princeton Conference on Information Sciences and Systems, Princeton University, March 1974. Accepted for publication in IEEE Trans. Automat. Contr.

[13] E. Wong, Stochastic Processes in Information and Dynamical Systems, McGraw-Hill Book Company, New York, 1971.

[14] H. Kushner, Stochastic Stability and Control, Academic Press, New York 1967.

[15] H. Kushner, Introduction to Stochastic Control, Holt, Rinehart and Winston, Inc., New York, 1971.

[16] I. Ia. Kats and N. N. Krasovskii, "On the stability of systems with random parameters", PMM, vol. 24, No. 5, pp. 809-823, 1960.

[17] H. Kushner, "Converse theorems for stochastic Lyapunov functions", J. SIAM Contr., ser. A, vol. 5, pp. 228-233, 1967.

[18] J. E. Bertram and P. E. Sarachik, "Stability of circuits with randomly time-varying parameters", IRE Trans. on Circuit Theory, vol. 6, pp. 260-270, 1959.

[19] H. Bunke, "Stabilität bei stochastischen Differentialgleichungssystemen", ZAMM, vol. 43, pp. 63-70, 1963.

[20] F. R. Gantmacher, The Theory of Matrices, Vol. II, Chelsea Publishing Co., New York, 1964.

[21] R. Bellman, Introduction to Matrix Analysis, McGraw-Hill Book Co., New York, 1967.

[22] L. Metzler, "Stability of multiple markets: the Hicks conditions", Econometrica, vol. 13, pp. 277-292, 1945.

[23] P. A. Samuelson, The Foundations of Economics Analysis, Harvard Univ. Press, Cambridge, Mass., 1947.

[24] I. W. Sandberg, "Some theorems on the dynamic response of nonlinear transistor networks", Bell Syst. Tech. J., vol. 48, pp. 35-54, 1969.

[25] J. Quirk and R. Saposnik, Introduction to General Equilibrium Theory and Welfare Economics, McGraw-Hill, New York, 1968.

[26] D. D. Siljak, "Competitive economic systems: stability, decomposition and aggregation", Proc. 1973 IEEE Conf. on Decision and Control, San Diego, 1973, pp. 265-275.

[27] R. M. May, "Stability in multispecies community models", Math. Biosciences, vol. 12, pp. 59-79, 1971.

This work was supported by the National Science Foundation under grant GK-33739 and by the Engineering Research Institute, Iowa State University.

TOLERANCE OF DECENTRALLY OPTIMAL CONTROLLERS TO NONLINEARITY AND COUPLING

STEIN WEISSENBERGER*
Ames Research Center, NASA, Moffett Field, Calif. 94035

ABSTRACT

A large system is represented as a collection of interconnected subsystems, each locally controlled by a linear-quadratic optimal control law based on a constant, linear subsystem model. Unmeasured states are estimated by observers. Sufficient conditions are found for the global asymptotic stability of the composite system with controller nonlinearity, modeling uncertainty, and nonlinear, time-varying coupling in state and control between subsystems. Controllers are permitted to saturate, giving rise to nonglobal stability properties, which are also investigated.

1. INTRODUCTION

Consider a large-scale system to be controlled by decentral, linear-optimal controllers: The large system is decomposed into subsystems, each controlled on the basis of its state alone; control laws are synthesized via linear, constant-coefficient models of subsystems as constant, linear-quadratic regulators. Unmeasured subsystem states are estimated by dynamic observers based on inaccurate process models. Observers may have knowledge of subsystem interconnections that are nonlinear, time-varying functions of subsystem states and controls. We wish to establish sufficient conditions for asymptotic stability of the composite system with controller nonlinearity and the various couplings and uncertainties.

Several recent papers [1-4] have considered related problems. Some [1,2] emphasized the stabilizability of large linear systems. In Ref. 3, local controllers were designed to stabilize a class of nonlinear, time-varying systems. In Ref. 4, the structural variations were classified under which a large dynamic system preserves asymptotic stability. In contrast, in our work we emphasize, at the same time, local optimality, nonlinearity, dynamic state estimation with modeling errors, structural variations, and nonglobal stability properties. The nonglobal aspect of the analysis gains significance because schemes for synthesizing stabilizing decentral controllers may depend on large feedback gains [3], and hence saturation effects become important.

Our approach to the problem is via the aggregation and comparison methods of stability analysis [4-6]. Several features of the work, combined controller nonlinearity and dynamic observers, have also received attention in the past for single subsystems [7,8]. Also, the spirit (and some of the subsystem analysis) of the present work is close to that of Ref. 7 in its discussion of the sensitivity of linear optimal regulators to nonlinearity.

The main problem is formulated in section 2 with a detailed system description. In section 3 and 4, the local control law and observer are specified. Section 5 contains the main results. Finally, section 6 states conclusions and gives suggestions for future work.

2. SYSTEM DESCRIPTION

Consider a large-scale system controlled decentrally through s local controllers u_i, each viewed as controlling a particular subsystem whose state is x_i, $i = 1,2, \ldots, s$. The system equations are

---

*NRC Senior Research Associate on leave from University of Santa Clara.

$$\dot{x}_i = A_i x_i + B_i\phi_i[u_i(\hat{x}_i),t] + \sum_{j=1}^{s} B_{ij}(x_j,t)\phi_j[u_j(\hat{x}_j),t] + \sum_{j=1}^{s} g_{ij}(x_j,t) \qquad (1)$$

$$y_i = H_i^T x_i \,, \qquad i = 1,2, \ldots, s \qquad (2)$$

where $\hat{x}_i$ is the estimate of x_i and y_i is the measured output of each subsystem. The system (A_i, B_i, H_i^T) is assumed to be completely controllable and completely observable. The estimates $\hat{x}_i$ are assumed to be obtained by a full-order observer described by*

$$\mathring{\hat{x}}_i = A_i\hat{x}_i + B_i\phi_i[u_i(\hat{x}_i),t] + \sum_{j=1}^{s} \hat{B}_{ij}(\hat{x}_j,t)\phi_j[u_j(\hat{x}_j),t] + \sum_{j=1}^{s} \hat{g}_{ij}(\hat{x}_j,t)$$

$$+K_{oi}^T(y_i - H_i^T\hat{x}_i) \qquad (3)$$

The dimensions of the vectors x_i, $\hat{x}_i$ are n_i, ϕ_i, u_i are m_i, y_i is ℓ_i, g_{ij}, $\hat{g}_{ij}$ are n_i, and A_i, B_i, B_{ij}, $\hat{B}_{ij}$, H_i, and K_{oi} are matrices of appropriate dimensions. A_i, B_i, K_{oi}, and H_i are constant. The matrices B_{ij}, $i \neq j$, represent the effects of control coupling between subsystems; B_{ii} represents ignored nonlinearity or uncertainty in our knowledge of B_i, the vectors $g_{ij}(x_j, t)$, $i \neq j$, represent the effects of state coupling between subsystems, and the function $g_{ii}(x_i,t)$ is the remnant in the subsystem description after extraction of the constant, linear representation $A_i x_i$, and results from nonlinearity or modeling errors in the representation of A_i. Thus, ϕ_i, B_{ij}, and g_{ij} can be used to represent nonlinearity, coupling, and uncertainty in the description of the overall systems. The observer design is based on assumed perfect knowledge of $\phi_i[u_i(\hat{x}_i),t]$ and on estimates $\hat{B}_{ij}$ and $\hat{g}_{ij}$ of B_{ij} and g_{ij}, respectively.**

Subtracting Eq. (1) from (3) yields

$$\dot{\tilde{x}}_i = C_i\tilde{x}_i + \sum_{j=1}^{s} \tilde{B}_{ij}(x_j,\tilde{x}_j,t)\phi_j[u_j(\hat{x}_j),t] + \sum_{j=1}^{s} \tilde{g}_{ij}(x_i,\tilde{x}_j,t) \qquad (4)$$

where

$$\tilde{x}_i \equiv \hat{x}_i - x_i \qquad (5)$$

is the error in the estimate of the ith state,

$$C_i \equiv A_i - K_{oi}^T H_i^T \qquad (6)$$

and

$$\tilde{B}_{ij}(x_j,\tilde{x}_j,t) \equiv \hat{B}_{ij}(\hat{x}_j,t) - B_{ij}(x_j,t) \qquad (7)$$

$$\tilde{g}_{ij}(x_j,\tilde{x}_j,t) \equiv \hat{g}_{ij}(\hat{x}_j,t) - g_{ij}(x_j,t) \qquad (8)$$

are (modeling) errors in the estimates of B_{ij} and g_{ij}, respectively. The state of the *overall* system (Eqs. (1), (4), and (5)) is defined as

---

*Subsequently, the annotation "$i = 1,2,\ldots, s$" is omitted from equations and variable citations where it is clearly implied.

**Imperfect knowledge of ϕ_i may be accounted for in simple and obvious extensions of the subsequent analysis.

$$\bar{x}^T \equiv \left[x_1^T, x_2^T, \ . \ . \ ., x_s^T, \tilde{x}_1^T, \tilde{x}_2^T, \ . \ . \ . \ , \tilde{x}_s^T\right] \tag{9}$$

with the actual and error states collectively denoted by, respectively,

$$\left.\begin{array}{l} x^T \equiv \left[x_1^T, \ x_2^T, \ . \ . \ ., \ x_s^T\right] \\ \\ \tilde{x}^T \quad \left[\tilde{x}_1^T, \ \tilde{x}_2^T, \ . \ . \ ., \ \tilde{x}_s^T\right] \end{array}\right\} \tag{10}$$

The state space of the overall system is denoted by $\bar{X}$ and that associated with the states x_i, x, $\tilde{x}_i$, $\tilde{x}$ by X_i, X, $\tilde{X}_i$, and $\tilde{X}$, respectively.

At the outset, we stipulate that the system (Eqs. (1), (4), (5)) has an equilibrium at $\bar{x} = 0$, and functions ϕ_i, g_{ij} satisfy $\phi_i(0,t) = 0$, $g_{ij}(0,t) = 0$, $t \in T$, where $T \equiv \{t: \ t \geq 0\}$.

Our problem will be to describe the stability properties of the equilibrium of system Eqs. (1), (4), and (5) based on characterizations of the local control laws u_i, the presence of observers, the amount of modeling errors $\tilde{B}_{ij}$, $\tilde{g}_{ij}$, and the nature of the functions ϕ_i, B_{ij}, and g_{ij}. In the next section, we begin by focusing on the design of a locally optimal controller for a subsystem based on perfect measurements of its state.

3. SUBSYSTEM ANALYSIS WITH PERFECT MEASUREMENTS

The *uncoupled* ith subsystem is defined by

$$\dot{x}_i = A_i x_i + B_i \phi_i[u_i(x_i),t] \tag{11}$$

where x_i is an n_i-dimensional state vector, $u_i(x_i)$ is an m_i-dimensional control vector function of x_i, ϕ_i is an m_i-dimensional vector function of u_i and t, and A_i and B_i are constant $n_i \times n_i$ and $n_i \times m_i$ matrices, respectively. Consider the synthesis of the control law $u_i(x_i)$ for Eq. (11) based on the minimization of the performance index

$$W_i = \int_0^\infty e^{2\alpha_i t}\left(x_i^T Q_i x_i + u_i^T R_i u_i\right) dt \tag{12}$$

Šiljak [9] introduced such exponentially weighted quadratic performance measures in the problem of performance optimization subject to "stability constraints," and Anderson and Moore [7] made extensive use of the form (Eq. (12)) in the context of the problem of control of systems described by Eq. (11). The subsequent global stability analysis follows their work closely.

Assume that α_i, Q_i, R_i are constant, $R_i = \text{diag}\ \{r_{ij}\}$, $r_{ij} > 0$, $j = 1,2, \ . \ . \ ., m_i$, (A_i, B_i) is completely controllable and $(A_i, Q_i^{1/2})$ is completely observable [7]. Consider first the (nominal, linear, constant) case $\phi_{ij}(u_i) = u_{ij}$, $j = 1,2, \ . \ . \ ., m_i$. Then the well-known solution $u_i(x_i)$ to the problem of the minimization of Eq. (12) is

$$u_i(x_i) = -K_i^T x_i \tag{13}$$

where

$$K_i = P_i B_i R_i^{-1} \tag{14}$$

and P_i is the positive-definite solution of the Ricatti equation

$$P_i(A_i + \alpha_i I) + (A_i^T + \alpha_i I)P_i - P_i B_i R_i^{-1} B_i^T P_i + Q_i = 0 \tag{15}$$

Now consider the use of the control law Eq. (13) in the nonlinear system Eq. (11). Noting that the optimal cost function at any time t for the linear problem is given by

$$W_i[x_i(t)] = x_i(t)^T P_i x_i(t) e^{2\alpha_i t} \tag{16}$$

we consider the time derivative of the function

$$V_i(x_i) = x_i^T P_i x_i \tag{17}$$

on solutions of Eq. (11), to obtain after some manipulation and utilization of Eqs. (13)-(15),

$$\dot{V}_i \Big|_{(11)} \le -2\alpha_i V_i \tag{18}$$

provided that $\phi_i(u_i)$ satisfies

$$u_i^T R_i \phi_i(u_i,t) > (1/2) u_i^T R_i u_i \tag{19}$$

A sufficient condition to satisfy Eq. (19) is that $\phi_{ij}(u_i,t) = \phi_{ij}(u_{ij},t)$ and

$$\phi_{ij}(u_{ij},t)/u_{ij} \ge 1/2 \ , \qquad j = 1,2, \ . \ . \ ., m_i \tag{20}$$

Thus, $V_i(x_i)$ is a Liapunov function for the subsystem Eq. (11), which establishes global asymptotic stability of Eq. (11), with the property Eq. (18), provided the functions $\phi_{ij}(u_{ij},t)$ belong to the sectors Eq. (20).

Consider the case in which the conditions of Eq. (20) fail to be satisfied everywhere. Specifically, assume that Eq. (20) is satisfied for a finite, local range of values of u_{ij},

$$-\eta_{ij} \le u_{ij} \le \mu_{ij} \ , \qquad j = 1,2, \ . \ . \ ., m_i \tag{21}$$

where $\eta_{ij} > 0$, $\mu_{ij} > 0$, $j = 1,2, \ . \ . \ ., m_i$. Such conditions would arise, for example, if each nonlinearity ϕ_{ij} were subject to saturation. It is readily shown that an estimate of a region Ω_i of asymptotic stability of Eq. (11) is then given by

$$\Omega_i = \{x_i \ : \ V_i(x_i) < \gamma_i\} \tag{22}$$

where

$$\gamma_i = \underset{1 \le j \le m_i}{\text{minimum}} \left\{ \left(k_{ij}^T P_i^{-1} k_{ij} \right)^{-1} \min\left(\eta_{ij}^2, \ \mu_{ij}^2 \right) \right\}$$

and the n_i-dimensional vector k_{ij} is the jth column of K_i. (See Refs. 10 and 11 for a similar calculation for the case $m_i = 1$.)

4. OBSERVER

We now elaborate on the observer described in Eqs. (3) through (8). Note first that this particular full-order structure is not necessary and that other structures [8] may be adapted to the subsequent analysis. For our purposes, we need only note that complete observability of (A_i, H_i^T) permits the arbitrary selection of the n_i eigenvalues λ of C_i through an appropriate choice of K_{oi}. They may, for example, be so chosen that, for any number $\beta_i > 0$,

$$\mathrm{Re}[\lambda_j(C_i)] < -\beta_i , \quad j = 1,2, \ldots, n_i \tag{23}$$

Consider now the observer equations (4) for the case of *perfect modeling:*

$$\dot{\tilde{x}}_i = C_i\tilde{x}_i \tag{24}$$

Because of Eq. (23), a Liapunov function $\tilde{V}_i$,

$$\tilde{V}_i = \tilde{x}_i^T\tilde{P}_i\tilde{x}_i \tag{25}$$

exists for the *observer* subsystem (24) so that the derivative of Eq. (25) on solutions of (24) is bounded by

$$\dot{\tilde{V}}_i\Big|_{(24)} < -2\beta_i\tilde{V}_i \tag{26}$$

where $\tilde{P}_i$ is the positive-definite solution of

$$\tilde{P}_i(C_i + \beta_i I) + (C_i^T + \beta_i I)\tilde{P}_i + \tilde{Q}_i = 0$$

where $\tilde{Q}_i$ is an arbitrary n_i-dimensional positive-definite matrix.

5. SYSTEM STABILITY ANALYSIS

Now reconsider the system equations (1), (4), and (5) in light of the development of the local control law (Eq. (13)). Combining these equations yields

$$\begin{aligned}
\dot{x}_i &= A_i x_i + B_i\phi_i\left(-K_i^T x_i, t\right) + \sum_{j=1}^{s} B_{ij}(x_j,t)\phi_j\left(-K_j^T x_j, t\right) \\
&\quad + \sum_{j=1}^{s} B_{ij}(x_j,t)f_j(x_j,\tilde{x}_j) + \sum_{j=1}^{s} g_{ij}(x_j,t) \\
\mathring{\tilde{x}}_i &= C_i\tilde{x}_i + \sum_{j=1}^{s} \tilde{B}_{ij}(x_j,\tilde{x}_j,t)\phi_j\left(-K_j^T x_j, t\right) + \sum_{j=1}^{s} \tilde{B}_{ij}(x_j,\tilde{x}_j,t)f_j(x_j,\tilde{x}_j) \\
&\quad + \sum_{j=1}^{s} \tilde{g}_{ij}(x_j,\tilde{x}_j,t)
\end{aligned} \tag{27}$$

where

$$f_i(x_i,\tilde{x}_i,t) \equiv \phi_i\left[-K_i^T(x_i + \tilde{x}_i),t\right] - \phi_i\left(-K_i^T x_i, t\right) \tag{28}$$

Now assume that $\phi_i(u_i,t)$, $B_{ij}(x_j,t)$, and $g_{ij}(x_j,t)$ are such that

$$\left|\sum_{j=1}^{s} g_{ij}(x_j,t)\right| \leq \sum_{j=1}^{s} \rho_{1_{ij}} |x_j| \tag{29}$$

$$\left|\sum_{j=1}^{s} B_{ij}(x_j,t)\phi_j\left(-K_j^T x_j,t\right)\right| \le \sum_{j=1}^{s} \rho_{2_{ij}} |x_j| \tag{30}$$

$$\left|\sum_{j=1}^{s} B_{ij}(x_j,t) f_j(x_j,\tilde{x}_j,t)\right| \le \sum_{j=1}^{s} \rho_{3_{ij}} |\tilde{x}_j| \tag{31}$$

and that $\tilde{g}_{ij}(x_j,\tilde{x}_j,t)$ satisfy

$$\left|\sum_{j=1}^{s} \tilde{g}_{ij}(x_j,\tilde{x}_j,t)\right| \le \sum_{j=1}^{s} \left(\tilde{\rho}_{1_{ij}} |x_j| + \tilde{\rho}_{4_{ij}} |\tilde{x}_j|\right) \tag{32}$$

It is assumed that $\tilde{B}_{ij}(x_j,\tilde{x}_j,t)$ is such that precisely the same inequalities as Eqs. (30) and (31) hold if B_{ij} is replaced by $\tilde{B}_{ij}$ and $\rho_{2_{ij}}, \rho_{3_{ij}}$ are replaced by $\tilde{\rho}_{2_{ij}}, \tilde{\rho}_{3_{ij}}$. The bounds of Eqs. (30) and (31) will hold, for example, if the elements of $B_{ij}(x_j,t)$ are bounded for $t \in T$, and $\phi_i(u_i)$ satisfy a uniform Lipschitz condition in u_i, uniformly in t. The functions $\phi_i(u_i,t)$ are further assumed to satisfy Eq. (20).

If the subsystem Liapunov functions

$$v_i(x_i) \equiv [V_i(x_i)]^{1/2}, \quad \tilde{v}_i(\tilde{x}_i) \equiv [\tilde{V}_i(\tilde{x}_i)]^{1/2} \tag{33}$$

are introduced, we can establish from Eqs. (27)-(33) the comparison equations [4]

$$\dot{\bar{v}} \le A\bar{v}, \quad t \in T, \quad x \in \Omega \times X \tag{34}$$

where

$$\bar{v}^T \equiv [v^T, \tilde{v}^T]$$

$$v^T \equiv [v_1, v_2, \ldots, v_s]$$

$$\tilde{v} \equiv [\tilde{v}_1, \tilde{v}_2, \ldots, \tilde{v}_s]$$

$$\Omega \equiv \Omega_1 \times \Omega_2 \times \ldots \times \Omega_s$$

and the elements of A are given by

$$\left.\begin{aligned}
a_{ij} &= -\delta_{ij}\alpha_i + \left(\rho_{1_{ij}} + \rho_{2_{ij}}\right)\lambda_i^{-1/2}\Lambda_i\lambda_j^{-1/2}, \quad i,j = 1,2,\ldots,s \\
&= \rho_{3i(j-s)}\lambda_i^{-1/2}\Lambda_i\tilde{\lambda}_{j-s}^{-1/2}, \quad \begin{cases} i = 1,2,\ldots,s \\ j = s+1,s+2,\ldots,2s \end{cases} \\
&= [\tilde{\rho}_{1(i-s)j} + \tilde{\rho}_{2(i-s)j}]\tilde{\lambda}_{i-s}^{-1/2}\tilde{\Lambda}_{i-s}\lambda_j^{-1/2}, \quad \begin{cases} i = s+1,s+2,\ldots,2s \\ j = 1,2,\ldots,s \end{cases} \\
&= -\delta_{ij}\beta_i + [\tilde{\rho}_{3(i-s)(j-s)} + \tilde{\rho}_{4(i-s)(j-s)}]\tilde{\lambda}_{i-s}^{-1/2}\tilde{\Lambda}_{i-s}\tilde{\lambda}_{j-s}^{-1/2}, \\
&\qquad\qquad i,j = s+1,s+2,\ldots,2s
\end{aligned}\right\} \tag{35}$$

where δ_{ij} is the Kronecker delta and λ_i, Λ_i and $\tilde{\lambda}_i, \tilde{\Lambda}_i$ are the minimum and maximum eigenvalues, respectively, of the matrices $P_i, \tilde{P}_i$.

The matrix A may be partitioned as

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}. \qquad (36)$$

where the A_{ij} are $s \times s$. Note that A_{12} and A_{21} are non-negative and that A_{11} and A_{22} have non-negative off-diagonal elements. Thus A is a *Metzler* matrix and certain stability properties of Eq. (27) depend on whether A is Hurwitz, i.e., whether A is a *Hicks* matrix [4,6]. We immediately obtain the following.

Theorem 1. Global Stability

Suppose that $\Omega = X_1 \times X_2 \times \ldots \times X_s$ (each subsystem (Eqs. (11),(13)) is globally asymptotically stable with $\gamma_i = \infty$, $i = 1,2, \ldots, s$). Then the system (Eq. (27)) is globally asymptotically stable if A is Hicks, i.e., if

$$a_{11} < 0 \; , \quad \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} > 0 \; , \ldots , (-1)^{2s} \det A > 0 \qquad (37)$$

Clearly, Eq. (37) will hold if $\alpha_i, \beta_i > 0$ and if coupling and modeling uncertainties are sufficiently small, i.e., the coefficients $\rho, \tilde{\rho}$ are small enough. This result is valid for the entire indicated class of functions, i.e., for B_{ij}, g_{ij}, $\tilde{B}_{ij}$, $\tilde{g}_{ij}$, and ϕ_i that satisfy Eqs. (29)-(32) and (20). Thus, when subsystem controllers are designed to establish Eq. (37), a "structural stability" property occurs with respect to functional perturbations within these classes, including, for example, arbitrary connections and disconnections of subsystems through B_{ij} and g_{ij} perturbing the controlled subsystems directly as well as changing (or introducing) modeling-error effects in the observers, all provided the conditions (Eqs. (29)-(32) and (20)) remain satisfied: the controllers are in this sense "robust."

Somewhat stronger results are obtained if the observer is based on an exact model of the process (Eq. (1)), i.e., when $\tilde{B}_{ij} \equiv 0$, $\tilde{g}_{ij} \equiv 0$.* This assumption is made throughout the remainder of this paper. Then, $A_{12} = 0$ and $A_{22} = -\mathrm{diag}(\beta_i)$; thus if A_{11} is Hicks, then A is Hicks and Theorem 2 immediately follows.

Theorem 2. Global Stability - Exact Model

Suppose that $\Omega = X_1 \times X_2 \times \ldots X_s$ and that $\tilde{B}_{ij} \equiv 0$, $\tilde{g}_{ij} \equiv 0$. Then system (27) is globally asymptotically stable if A_{11} is Hicks. This result is a generalization of the well-known phenomena for perfectly modeled single observers, established by Luenberger [8] for $s = 1$ and $g_{11} \equiv 0$.

Consider the situation when each subsystem in Eq. (11), $i = 1,2,\ldots,s$, is guaranteed to be asymptotically stable only in a region Ω_i given by Eq. (22). We wish to estimate the region of asymptotic stability (RAS) Θ in state space $\bar{X}$ where the entire interconnected system (27) is asymptotically stable. We rely on the results of Ref. 6.

From Eqs. (22) and (33),

$$\Omega_i = \{x_i \;:\; v_i(x_i) < v_{oi}\} \qquad (38)$$

---

*Similar results can be obtained without the strong assumption $\tilde{g}_{ij} \equiv 0$, i.e., with $\hat{g}_{ij} = g_{ij}(\hat{x}_j, t)$.

where $v_{oi} \equiv \gamma_i^{1/2} < \infty$. First consider the case of perfect measurements of the state x, i.e., set $\tilde{x} \equiv 0$, $t \in T$, in Eq. (1) and consider the reduced set of equations in x only. Denote under this condition the RAS for no control coupling by Ω, where clearly

$$\Omega = \Omega_1 \times \Omega_2 \times \ldots \times \Omega_s \tag{39}$$

Now denote by Ω^* an estimate of this RAS when coupling B_{ij}, g_{ij} is introduced. From Theorem 5 of Ref. 6, $\Omega^* = \Omega$ if A_{11} is Hicks and

$$A_{11}v_o < 0\text{*}\ , \qquad v_o^T \equiv [v_{o1}, v_{o2}, \ldots, v_{os}] \tag{40}$$

We now give bounds on initial observer states, for given classes of nonlinearity and coupling, so that asymptotic stability prevails throughout $\Omega^* = \Omega$.

Theorem 3. Region of Stability

Suppose that $\tilde{B}_{ij} \equiv 0$, $\tilde{g}_{ij} \equiv 0$. Then there exist nonempty regions $\tilde{\Omega}_i \subseteq \tilde{X}_i$, such that the RAS of Eq. (27) is estimated by

$$\Theta = \Omega \times \tilde{\Omega}_1 \times \tilde{\Omega}_2 \times \ldots \times \tilde{\Omega}_s$$

if A_{11} is Hicks and if $A_{11}v_o < 0$. The regions $\tilde{\Omega}_i$ are given by

$$\tilde{\Omega}_i = \{\tilde{x}_i \ : \ \tilde{v}_i(\tilde{x}_i) < \tilde{v}_{oi}\} \tag{41}$$

where

$$\tilde{v}_o > 0\ , \quad A_{11}v_o + A_{12}\tilde{v}_o \leq 0\ ; \quad \tilde{v}_o^T \equiv \left[\tilde{v}_{o1}, \ldots, \tilde{v}_{os}\right] \tag{42}$$

The proof follows directly from Theorem 5 of Ref. 6, together with the fact that $A_{22} = -\mathrm{diag}(\beta_i)$, $\beta_i > 0$. For no control coupling ($B_{ij} = 0$), $\tilde{v}_o$ is readily computed as

$$\tilde{v}_o = -\mathrm{diag}\left(\rho_{3ii}^{-1}\lambda_i\Lambda_i^{-1}\right)A_{11}v_o \tag{43}$$

Note that $\tilde{v}_o$ and the regions (Eq. (41)) determine bounds on initial measurement errors:

$$|\hat{x}_i - x_i| = |\tilde{x}_i| < \tilde{\lambda}_i^{-1/2}\tilde{v}_{oi} \tag{44}$$

6. CONCLUSIONS

This paper has shown that certain stability properties are preserved by a large-scale system under locally optimal control when it is subject to a variety of nonlinear, time-varying imperfections in controller characteristics, coupling between subsystems, and modeling errors, provided nonlinearity, coupling, and errors are sufficiently small. For perfect modeling and control saturation, bounds were formed on initial errors in state estimates so that coupling and nonlinearity does not effect the RAS in the process states.

---

*The notation $a < b$ means that $a_i < b_i$ for all i; $a \leq b$ means that $a_i \leq b_i$ for all i.

Estimation of transient response and performance degradation apparently is a straightforward extension of this work. Equally natural (but not equally straightforward) would be to consider the converse or *synthesis* problem: Given a class description of nonlinearity and coupling, how does one design local controllers to achieve desired stability properties of the overall system? Davison [3] presented a solution to this solution for a certain class of systems by another method, and it is natural to inquire whether solutions can also be found within the context of the present work.

Although a definitive answer is not presently available, preliminary indications are that, in general, a näive application of the machinery of this paper to the synthesis problem is not possible. The most obvious potential method for stabilizing the overall system, choosing the numbers α_i sufficiently large, can be shown by counterexample to fail in this objective. More definitive work on this aspect of the problem is clearly desirable, although it would not be surprising that strict attention to local optimality (even with a degree-of-stability constraint) is sometimes contradictory to the objective of overall stabilization.

REFERENCES

[1] M. Aoki, "On feedback stabilizability of decentralized dynamic systems," Automatica, vol. 8, pp. 163-173, March 1972.

[2] S. Wang and E. J. Davison, "On the stabilization of decentralized control systems," IEEE Trans. Autom. Contr., vol. AC-18, pp. 473-478, Oct. 1973.

[3] E. J. Davison, "The decentralized stabilization and control of a class of unknown non-linear time-varying systems," Automatica, vol. 10, pp. 309-316, May 1974.

[4] D. D. Šiljak, "Stability of large-scale systems under structural perturbations," IEEE Trans. Syst. Man. Cybernetics, vol. SMC-2, pp. 657-663, Nov. 1972.

[5] F. N. Bailey, "The application of Lyapunov's second method to interconnected systems," J. SIAM Contr., vol. 3, pp. 443-462, no. 3, 1966.

[6] S. Weissenberger, "Stability regions of large-scale systems," Automatica, vol. 9, pp. 653-663, Nov. 1973.

[7] B. D. O. Anderson and J. B. Moore, Linear Optimal Control, Prentice-Hall, Englewood Cliffs, N. J., 1971.

[8] D. G. Luenberger, "Observers for multivariable systems," IEEE Trans. Autom. Contr., vol. AC-11, pp. 190-197, April 1966.

[9] D. D. Šiljak, "Square-error minimization with stability constraints," IEEE Trans. Autom. Contr., vol. AC-11, pp. 589-591, Oct. 1968.

[10] S. Weissenberger, "Application of results from the absolute stability problem to the computation of finite stability domains," IEEE Trans. Autom. Contr., (Corresp.), vol. AC-13, pp. 124-125, Feb. 1968.

[11] J. A. Walker and H. N. McClamroch, "Finite regions of attraction for the problem of Luré," Int. J. Contr., vol. 6, pp. 331-336, 1967.

REMARKS ON ALGEBRAIC DECOMPOSITION OF PERMUTATION AUTOMATA

VIJAY B. AGGARWAL
Department of Mathematics
University of Vermont
Burlington, Vermont 05401

DAVID E. MULLER
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois

ABSTRACT

The constructive half of Krohn-Rhodes decomposition theorem uses irreducible semiautomata that are complete. This assumption often requires the basic building semiautomata to be more complex than necessary or desirable. Considering the length of the input strings we obtain the relationship of a semiautomaton to its completion for group-machines. It is shown that a cyclic-machine with a single input is strictly weaker than its completion. However, for a cyclic-machine with two distinct inputs, the completion can be generated using series-parallel connections. For a set $\mathcal{M}$ of cyclic-machines, just one machine with two distinct inputs is sufficient to generate the complete machines for all members of $\mathcal{M}$ in the series parallel closure of $\mathcal{M}$. Using the above results a stronger constructive result is obtained for semiautomata whose semigroups are solvable groups.

1. INTRODUCTION

The main problem in the algebraic theory of series-parallel decomposition of automata can be described briefly as follows: Given a semiautomaton A and a set $\mathcal{M}$ of semiautomata, under what conditions it is possible to construct a semiautomaton B by taking series-parallel connections of the semiautomata in $\mathcal{M}$, such that B has the capability of A?
Ideally we would like to have the following:

(i) The semiautomata in $\mathcal{M}$ are as simple as possible under some criteria.
(ii) The semiautomaton A has the capability of all the semiautomata in $\mathcal{M}$ that are actually used in the construction of A.

One version of the Krohn-Rhodes Theorem [2] is :

Theorem: A is in the series-parallel closure of $\mathcal{M}$, denoted by SP($\mathcal{M}$), if the following three conditions are satisfied.

(1) For every simple group K which divides the monoid M^A of A, there is a semiautomaton B in $\mathcal{M}$ such that K divides M^B.

(2) $W \in \mathcal{M}$, where W is the complete flip-flop (two-state identity-reset semiautomaton).

(3) All the semiautomata in $\mathcal{M}$ are complete, i.e. for A in $\mathcal{M}$, for each transformation in M^A of the states of A, there is a single input letter in the input alphabet of A, which produces the same transformation on the states of A.

If $A \in$ SP($\mathcal{M}$), then condition (1) necessarily holds but (2) and (3) may not. The flip-flop W may be required in the construction even when A does not have the capability of W. Also the question of whether one actually needs the complete machines required in (3) of the theorem is left open. The completion of a semiautomaton increases the number of inputs tremendously against the spirit (i) of the theory. Also the number of inputs available to a semiautomaton can make a significant difference in the capability of a semiautomaton as is shown in the following example.

Consider two permutation semiautomata A and B where A is a five-state semiautomaton with a single input which is a 5-cycle on the states and B is identical to A except that B has the identity input 'e' also. Both A and B have the same monoid namely the cyclic group of order 5. In SP(A), we can get only a mod 5 counter. However by connecting two copies of B in series-parallel, we can get a mod 25 counter. In fact we can construct 5^n counters for all integers in SP(B). Thus A and B have essentially distinct capabilities.

In this paper we study the relationship between a semiautomaton and its completion. We show that for any semiautomaton X whose semi-group is a cyclic group, and which has 2 input letters which are distinct transformations of the state set, then the complete machine $\overline{X}$ lies in SP(X).

Further, if $\mathcal{M}$ is a set of semiautomata whose semigroups are cyclic groups, then if one of these semiautomata has 2 inputs with distinct transformations, then we can get the complete semiautomaton on each of these semiautomata in SP($\mathcal{M}$). This result solves the case for a semiautomaton A whose monoid M^A is a solvable group. We can say exactly what components are needed for its construction. For each simple group K which divides M^A we need a semiautomaton whose monoid is K and one of these semiautomata must have 2 inputs with distinct transformations. This leaves the non-abelian simple group case to be solved.

2. TERMINOLOGY

Basically our notation follows Ginzburg [1]. Function f restricted to the set S is denoted by $f|S$. Cardinality of S is denoted by $|S|$.

A finite semiautomaton $A = (Q^A, I^A, f^A)$ consists of a finite set Q^A of states, a finite set I^A of inputs and a transition function $f^A: Q^A \times I^A \to Q^A$. The transition function f^A can be given by a set of functions from Q^A into Q^A indexed by the elements of I^A. The function from Q^A into Q^A indexed by 'a' in I^A is denoted by a^A. We shall omit the superscript A whenever it is clear from the context. f(q,a) is abbreviated to qa for q in Q and 'a' in I.

Let A, B be semiautomata. B is called a subsemiautomaton of A if $I^B \subset I^A$, $Q^B \subset Q^A$, and $a^B = a^A|Q^B$, the restriction of the function a^A to the set Q^B for all 'a' in I^B. B is called a non-trivial subsemiautomaton if $I^B = I^A$ and $|Q^A| > |Q^B| > 1$. B is called a homomorphic image of A if there are functions h: $Q^A \to Q^B$ and G: $I^B \to I^A$ such that h is onto and $h\,b^B = (G(b)^A)h$ for each 'b' in I^B, i.e., for all $q \in Q^A$ and all $b \in I^B$, $(h(q))b = h(qG(b))$. h is called homomorphism from A onto B. We say that B divides A (A coversB), in symbols $B|A$, if B is a homomorphic image of a subsemiautomaton of A.

A partition Π of the state set Q is called an admissible partition of the semiautomaton A if for all X in Π and 'a' in I, $Xa \subset Y$ for some Y in Π. The quotient semiautomaton $B = A/\Pi$ has the state set as Π, the input set as I^A and the transition function for 'a' in I^B is given by $X a^B = Y$ if $Xa \subset Y$. (Y is unique since Π is a partition.) Then B is a homomorphic image of A.

The semigroup S^A of a semiautomaton A is the transformation semigroup generated by $\{a^A : a \varepsilon I^A\}$ under composition of mappings. The monoid M^A of A is the semigroup S^A of A together with e^A, the identity on Q^A, if it is not already in the semigroup. Let S, T be semigroups. S divides T, in symbols $S|T$, iff S is a homomorphic image of subsemigroup of T. S is called a subgroup of T iff S is a subsemigroup of T and S is a group. For semiautomata A and B, if A divides B then M^A divides M^B. A semigroup S is called group-free if all its subgroups are trivial. A semiautomaton is called group-free if its monoid is group-free.

Let A be a semiautomaton. The completion of A is the semiautomaton $\bar{A}$ such that $Q^{\bar{A}} = Q^A$, $I^{\bar{A}} = M^A$ and for t in M^A, $t^{\bar{A}} = t^A$. A semiautomaton is called complete if it equals its completion. If S is a semigroup, the standard semiautomaton A(S) of the semigroup is (S, S, f) with $f(s_1, s_2) = s_1 s_2$ for s_1, s_2 in S. The standard semiautomaton of a monoid is complete.

Given a connecting function $w: Q^A \times I^A \rightarrow I^B$, the series-parallel connection or the cascade product $(A^o B)_w$ is the semiautomaton with state set $Q^A \times Q^B$, inputs I^A and the transitions given by $(p,q)\, a = (pa, q(w(p,a)))$ for (p,q) in $Q^A \times Q^B$ and a in I^A. The parallel connection $A \times B$ is the semiautomaton with the state set $Q^A \times Q^B$, input set $I^A \times I^B$ and the transitions given by $(p,q)(a,b) = (pa, qb)$ for $(p,q) \varepsilon Q^A \times Q^B$ and $(a,b) \varepsilon I^A \times I^B$.

Let $G: I \rightarrow I^A$ be a mapping of two input alphabets. Given a semiautomaton $A = (Q, I^A, f)$, then GA is defined to be the semiautomaton (Q, I, f') where $f'(q,a) = f(q, G(a))$ for all q in Q and a in I. GA is a homomorphic image of A or $GA|A$. The mapping G is used as input coding while making the series-parallel connections of semiautomata.

Let $\mathcal{M}$ be a set of semiautomata. The series-parallel closure $SP(\mathcal{M})$ of $\mathcal{M}$ is given by

$$SP(\mathcal{M}) = \bigcup \{SP_i(\mathcal{M}) : i = 1,2,3\ldots\}$$

where

$$SP_1(\mathcal{M}) = \mathcal{M}$$

and

$$SP_{i+1}(\mathcal{M}) = \left\{ \begin{array}{l} X: \text{where } X = A^o B \text{ or } X|A, \text{ for} \\ \quad A,B \text{ in } SP_i(\mathcal{M}) \end{array} \right\}$$

3. PRELIMINARY RESULTS

For each semiautomaton $A = (Q, I, f)$ we define the marked semigroup MS^A whose elements are pairs, the left coordinate is the length of the string t in I^* and the right coordinate is the mapping $t^A: Q^A \to Q^A$ and two elements are composed as follows:

$$(n,t_1^A) * (m,t_2^A) = (m+n, (t_1t_2)^A)$$

A length preserving homomorphism, (LP-homomorphism) $h: MS^A \to MS^B$ is described by specifying a homomorphism, also called h, from M^A to M^B (the right hand side of MS^A and MS^B respectively) which can be extended to the domain MS^A by preserving lengths, i.e.,

$$h(n,s_1) = (n,h(s_1)) \text{ for all } (n,s_1) \text{ in } MS^A$$

If MS^A and MS^B are two marked semigroups, then MS^A is said to divide MS^B, written $MS^A|MS^B$, iff there is a LP-homomorphism from a subsemigroup of MS^B onto MS^A.

Proposition 1: If A is a subsemiautomaton of B, then there is a LP-homomorphism from MS^B onto MS^A.

Proof: The homomorphism $h: M^B \to M^A$ given by $h(t^B) = t^A$ is well-defined ($t^A = t^B|Q^A$) and h is length preserving. Hence h can be extended to a LP-homomorphism from MS^B onto MS^A.

Proposition 2: Suppose $h: A \to B$ is a homomorphism of semiautomaton A onto B. Then h induces a length preserving homomorphism $H: MS \to MS^B$ (onto), where MS is a subsemigroup of MS^A.

Proof: The mapping $H: M \to M^B$ given by $H(t^A) = t^B$ is a homomorphism of M onto M^B where M is a submonoid of M^A. Also H is length preserving, hence H can be extended to a LP-homomorphism from MS onto MS^B where MS is a subsemigroup of MS^A.

Proposition 3: For semiautomata A,B, if $A|B$ then $MS^A|MS^B$.

Proof: Follows by combining propositions 1 and 2.

Lemma 1. Suppose A and B are semiautomata such that $MS^A = MS^B$. Then A can be covered by a cascade product of copies of B and an identity semiautomaton, i.e., A lies in $SP(B \cup U_0)$ where U_0 is a two-state identity semiautomaton.

Proof: Since $MS^A = MS^B$, we can find a one-to-one correspondence between the elements $\{(1,a_i) : a_i \in I^A\}$ of MS^A and the elements

$\{(1,b_i) : b_i \in I^B\}$ of MS^B of length one which gives rise to a one-to-one correspondence ϕ from I^B to I^A given by $b_i \leftrightarrow a_i$, after deliting identical inputs if necessary. This correspondence is chosen so that its extension to M^A and M^B gives an isomorphism $\phi: M^B \xrightarrow{\text{onto}} M^A$.

Now consider the cascade product C of an identity semiautomaton A_1 with the state set Q^A and n copies of B all acting in parallel where $|Q^B| = n$.

Clearly A_1 lies in $SP(U_o)$

$$Q^C = Q^A \times Q^B \times \ldots \times Q^B,\ I^C = I^B$$

The transition function $f^C: Q^C \times I^C \rightarrow Q^C$ is

$$(q_o^A, q_1, \ldots, q_n)\, b = (q_o^A, q_1 b, \ldots, q_n b)$$

where $q_o^A \in Q^A$, $q_i \in Q^B$, $i = 1, 2, \ldots, n$; and $b \in I^B$.

Let $Q^A = \{q_o^A, q_1^A, \ldots, q_k^A\}$ and $Q^B = \{1, 2, \ldots, n\}$

Take the subsemiautomaton D of C generated from the states $\{(q_o^A, 1, 2, \ldots, n), (q_1^A, 1, 2, \ldots, n), \ldots, (q_k^A, 1, 2, \ldots, n)\}$ under the input alphabet I^B.

Define a homormorphism $h: Q^D \rightarrow Q^A$ and $G: I^A \rightarrow I^D = I^B$ by setting $h(q_o^A, q_1, \ldots, q_n) = (q_o^A)(\phi(f))$ where G is the map of correspondence between I^A and I^B and f is the mapping determined by $(q_1, \ldots, q_n)$ according to the rule $f(i) = q_i$; thus $f: Q^B \rightarrow Q^B$ and since the semiautomaton D is generated by I^B, we have $f \in M^B$ and hence $\phi(f) \in M^A$ so that $(q_o^A)(\phi(f)) \in Q^A$. h and G are well defined maps.

Now h is onto, since the generating states in Q^D map onto Q^A. h is also a homomorphism for $(q_o^A, q_1, \ldots, q_n) \in Q^D$ and $a \in I^A$,

$$(h\,(q_o^A, q_1, q_2, \ldots, q_n))\ (a) = ((q_o^A)\ \phi(f))\ (a)$$

and $h\,((q_o^A, q_1, q_2, \ldots, q_n)\ G(a)) = h(q_o^A, q_1 G(a), q_2 G(a), \ldots, q_n (G(a)))$

and if the n-tuple $(q_1, q_2, \ldots, q_n)$ gives the mapping $f \in M^B$ then since ϕ is a homomorphism, the n-tuple $(q_1 G(a), q_2 G(a), \ldots, q_n G(a))$ gives the mapping $fG(a)$ in M^B.

$$h\,((q_o^A, q_1, \ldots, q_n)\ G(a)) = (q_o^A)(\phi(fG(a))) = (q_o^A)\ (\phi(f)a) =$$

$$((q_o^A)\ \phi\ (f))(a)$$

for the mappings G and ϕ restricted to I^B are inverses to each other and hence h gives the homomorphism from D onto A.

Therefore $A \varepsilon\ SP(B \cup U_o)$.

Remark: For any semiautomaton A which is not group-free, U_o lies in SP(A) and also U_o divides $\overline{A}$.

Lemma 2: Suppose A, B are semiautomata such that $MS^A | MS^B$. Then there is a cascade product of copies of B and an identity semiautomaton which covers A, i.e., $A \varepsilon\ SP(B \cup U_o)$.

Proof: Since $MS^A | MS^B$, there is a marked subsemigroup $MS' \subset MS^B$ such that ϕ: $MS' \to MS^A$ is a length preserving onto homomorphism of marked semigroups.

For each $a_i \varepsilon I^A$, $(1, a_i) \varepsilon MS^A$ and since ϕ is onto, there exists $(1, b_i)$ in MS' such that $\phi(1, b_i) = (1, a_i)$.

By making a suitable choice for each $a_i \varepsilon I^A$, we get the set $\{(1, b_i): b_i \varepsilon I^B\}$ MS' which is in one-to-one correspondence with the set $\{1, a_i): a_i \varepsilon I^A\}$ and this correspondence preserves the semigroup homomorphism between the right sides, i.e., $\phi: M' \to M^A$ is an onto homomorphism.

Take the subsemiautomaton C of B by restricting the input alphabet I^C to exactly those b_i in I^B which correspond to a_i in I^A.

Clearly $MS^C \subset MS'$, $M^C \subset M'$ and $\phi: M^C \to M^A$ is an onto mapping. Now consider a cascade product D of an identity semiautomaton A_1 with state set Q^A and n copies of C all acting in parallel where $|Q^C| = n$.

$$Q^C = Q^B;\ I^C \subset I^B$$

$$Q^D = Q^A \times Q^C \times Q^C \times \ldots \times Q^C,\ I^D = I^C$$

Transition function f^D: $Q^D \times I^D \to Q^D$ is

$$(q_o^A, q_1, q_2, \ldots, q_n)\, b = (q_o^A, q_1 b, q_2 b, \ldots, q_n b)$$

where $q_o^A \varepsilon\ Q^A$, $b \varepsilon\ I^D$, $q_i \varepsilon\ Q^B$ for i = 1, 2, 3, . . . , n.

Let $Q^A = \{q_o^A, q_1^A, \ldots, q_k^A\}$ and $Q^C = \{1, 2, \ldots, n\}$

Take the subsemiautomaton E of D generated from the states $\{(q_o^A, 1, 2, \ldots, n), (q_1^A, 1, 2, \ldots, n), \ldots, (q_k^A, 1, 2, \ldots, n)\}$ under the input alphabet $I^C = I^D$.

Define a homomorphism h: $Q^E \to Q^A$ and G: $I^A \to I^E$ by setting $h(q_o^A, q_1, \ldots, q_n) = (q_o^A)(\phi(f))$ where f in M^C is determined by the n-tuple $(q_1, q_2, \ldots, q_n)$ according to $f(i) = q_i$ so that $\phi(f)$ lies in M^A and hence $(q_o^A)(\phi(f))$ lies in Q^A.

Clearly h is onto Q^A and h is a homomorphism. Hence A divides D so that $A \varepsilon \; SP(B \cup U_0)$.

4. THE GROUP CASE

In the construction of a semiautomaton A using series-parallel connections, the Krohn-Rhodes decomposition theorem [2] uses complete semiautomata for each simple group K such that $K | M^A$. For each prime number p, the cyclic group of order p is a simple group. We first study the semiautomata whose monoids are cyclic groups of prime order with different input sets.

Theorem 1: Suppose $A = (Q^A, I^A, f^A)$ is a semiautomaton where $I^A = \{e,a\}$, $Q^A = \{0,1,2, \ldots, n-1\}$ such that for $\forall \; i \varepsilon Q^A$, $ie = i$ and $ia = i + 1 \pmod n$. Then the complete semiautomaton $\bar{A}$ on A lies in SP(A).

Proof: The construction of $\bar{A}$ from copies of A is carried out as follows. Let $I^B = \{e, b_1, b_2\}$ and using the two input codings G_1 and G_1'

$G_1 : I^B \to I^A$

$G_1' : I^B \to I^A$

	G_1	G_1'
e	e	e
b_1	a	e
b_2	e	a

defined by the table, form the parallel connection B of semiautomata G_1A and $G_1'A$ on I^B given by

$Q^B = \{(i,j) : 0 \leq i \leq n-1 \text{ and } 0 \leq j \leq n-1\}$ $I^B = \{e, b_1, b_2\}$;

and the transitions in B are

$(i,j)s = (iG_1(s), jG_1'(s))$; for $(i,j,) \varepsilon Q^B$, $s \varepsilon I^B$.

Consider the following partition Π_1 of Q^B

	Constant difference in coordinates
$p_0 = \{(0,0), (1,1), \ldots, (n-1,n-1)\}$	0
$p_1 = \{(1,0), (2,1), \ldots, (0, n-1)\}$	1
$p_2 = \{(2,0), (3,1), \ldots, (1, n-1)\}$	2
.	
.	
.	
$p_{n-1} = \{(n-1,0), (0,1), \ldots, (n-2, n-1)\}$	n-1

This partition Π_1 of Q^B is admissible for

(i) $p_i e = p_i$ for $0 \leq i \leq n-1$. ('e' leaves the "difference" unchanged)

(ii) $p_i b_1 = p_{i+1} \pmod{n}$ for the input b_1 is $\{\overset{a}{e}\}$ so that $(i,j)\, b_1 = (i+1,j)$. ('b_1' increases the "difference" by 1)

(iii) $p_i b_2 = p_{i-1} \pmod{n}$ for the input b_2 is $\{\overset{e}{a}\}$ so that $(i,j)\, b_2 = (i,j+1)$. ('b_2' decreases the "difference" by 1)

that is, for $\forall\, s \in I^B$ and $P \in \Pi$, there is a $Q \in \Pi$, such that $Ps \subset Q$.

Thus the quotient semiautomaton B/Π_1 has n states and inputs $\{e,b_1,b_2\}$ such that e is the identity on the state set, b_1 is an n-cycle and b_2 is the inverse cycle. Denote the semiautomaton B/Π_1 by C.

Next we form the parallel connection D of the semiautomata G_2C and $G_2'C$ on the input alphabet $I^D = \{e,c_1,c_2,c_3,c_4\}$ where the mappings are defined by the table:

$G_2 : I^D \to I^C$

$G_2' : I^D \to I^C$

	G_2	G_2'	Exponent
e	e	e	0
c_1	b_1	e	1
c_2	e	b_1	-1
c_3	b_1	b_2	2
c_4	b_2	b_1	-2

The following partition Π_2 of Q^D is admissible.

	Constant difference in coordinates.
$p_0 = \{(p_0,p_0), (p_1,p_1),\ldots,(p_{n-1}, p_{n-1})\}$	0
$p_1 = \{(p_1,p_0), (p_2,p_1),\ldots, (p_0, p_{n-1})\}$	1
$p_2 = \{(p_2,p_0), (p_3,p_1),\ldots, (p_1,p_{n-1})\}$	2
.	
.	
.	
$p_{n-1} = \{(p_{n-1},p_0), (p_0,p_1),\ldots , (p_{n-2}, p_{n-1})\}$	n-1

Π_2 is admissible for

when $s = e$, $p_i e = p_i$ for $0 \leq i \leq n-1$ (leaves the "difference" unchanged)

when $s = c_1$, the input is $\{b_i,e\}$ so that $(p_i,p_j)c_1 = (p_{i+1}, p_j)$; so the input increases the "difference" by 1, that is $p_i c_1 = p_{i+1} \pmod{n}$ for every i.

When $s = c_2$, the input is $\{e,b_1\}$ so that $(p_i,p_j)c_2 = (p_i,p_{j+1})$; so the input decreases the "difference" by 1, that is $p_i c_2 = p_{i-1} \pmod{n}$ for every i.

When $s = c_3$, the input is $\{b_1,b_2\}$ so that $(p_i,p_j)c_3 = (p_{i+1}, p_{j-1})$ so the input increases the "difference" by 2, that is $p_i c_3 = p_{i+2} \pmod n$ for every i.

When $s = c_4$, the input is $\begin{Bmatrix} b_2 \\ b_1 \end{Bmatrix}$ so that $(p_i,p_j)\, c_4 = (p_{i-1}, p_{j+1})$ so the input decreases the "difference" by 2, that is

$$p_i c_4 = p_{i-2} \pmod n \text{ for } 0 \leq i \leq n-1.$$

Thus the quotient semiautomaton $E = D/\Pi_2$ has $Q^E = \{p_0,p_1,\ldots, p_{n-1}\}$ and input alphabet $I^E = \{e, c_1, c_2, c_3, c_4\}$ such that e is the identity on the state set, c_1 is an n-cycle on Q^E, c_2 is the inverse cycle, $c_3 = (c_1)^2$, $c_4 = (c_2)^2 = (c_1)^{-2}$.

Proceeding in the above way, we can get the complete machine $\bar{A}$ on A in SP(A) since all the constructions made are allowable in the series-parallel closure.

Due to the limited space available, we present below a summary of the final results.

Theorem 2: Let A be a semiautomaton with input set $I^A = \{a,b\}$ such that M^A is a cyclic group and such that the transformations a^A and b^A induced by a and b on the state set of A are unequal. Then the completion $\bar{A} \in SP(A)$.

Theorem 3: Let $\mathcal{M}$ be a set of semiautomata such that for all $A \in \mathcal{M}$, M^A is a cyclic group and for some $B \in \mathcal{M}$, $I^B = \{b_1,b_2\}$ such that $b_1^B \neq b_2^B$. Then $\bar{A} \in SP(\mathcal{M})$ for all $A \in \mathcal{M}$.

1. For a semiautomaton A, such that $|Q^A| > 2$, $I^A = \{a\}$ and M^A is a cyclic group. Then $SP(A) = \left\{ \begin{array}{l} X : X \text{ is a semiautomaton dividing } AxAx\ldots.xA, \\ \text{the parallel connection of copies of } A. \end{array} \right\}$

2. For a semiautomaton A, such that $I^A = \{a,b\}$, $a^A \neq b^A$ and M^A is a cyclic group. Let $|M^A| = n = p_1^{e_1} \; p_2^{e_2} \; . \; . \; . \; . \; p_r^{e_r}$. Then $SP(A) = \left\{ \begin{array}{l} \text{Semiautomaton } B\text{: } M^B \text{ is a group and for } \forall \text{ simple group } K|M^B, \\ \qquad |K| = p_i,\ 1 \leq i \leq r \end{array} \right\}$

3. Let $\mathcal{M} = \{A_1,A,\ldots\ldots,A_t\}$ be a set of semiautomata such that for all i, $1 \leq i \leq t$, $I^{A_i} = \{a_i\}$ and M^{A_i} is a cyclic group. Let $|M^{A_i}| = n_i$.

$SP(\mathcal{M}) = \left\{ \begin{array}{l} B : B \text{ has a single input letter and } M^B \text{ is a group and} \\ \qquad |M^B| \text{ divides l.c.m. } (n_1, n_2, \ldots, n_t). \end{array} \right\}$

4. Let $\mathcal{M}$ be a set of semiautomata such that for all $A \in \mathcal{M}$, M^A is a cyclic group and for some $A \in \mathcal{M}$, A has 2 distinct inputs, then

$$SP(\mathcal{M}) = \left\{ \begin{array}{ll} B \;:\; & M^B \text{ is a group and for } \forall \text{ simple group } K|M^B,\ \exists\, A \in \mathcal{M} \\ & \text{such that } |K| \text{ is a prime dividing } |M^A|. \end{array} \right\}$$

These results tell us exactly what components are needed for the construction of a semiautomaton A whose monoid M^A is a solvable group. For each simple group K which divides M^A we need a semiautomaton whose monoid is K and one of these semiautomata must have 2 inputs with distinct transformations. To obtain similar results for an arbitrary permutation semiautomaton, we need to study the completions of semiautomata whose monoids are non-abelian simple groups.

References

[1] A.Ginzburg, "Algebraic Theory of Automata," Academic Press, 1968.

[2] K.Krohn and J. Rhodes, "Algebraic Theory of Machines, 1", Trans. Amer. Math. Soc. 116, (1965), 450-464.

[3] H.Zeiger, "Cascade Synthesis of finite state machines", Inf and Control 10, 419-433 (1967).

[4] V.Aggarwal, "Algebraic Decomposition of Automata", Tech. Rep. CS-1, Department of Mathematics and Computer Science, Colorado State University, January 1973.

A NEW ALGORITHM FOR DETERMINING THE ISOMORPHISM OF DIRECTED GRAPHS

LARRY E. DRUFFEL & DOUGLAS C. SCHMIDT
Systems and Information Science Program
Vanderbilt University, Nashville, Tennessee.

A new algorithm to test a pair of directed graphs for isomorphism is presented in this paper. The algorithm is complete in that it terminates providing an isomorphism or terminates indicating that no isomorphism exists.

DEFINITIONS

A directed graph, or digraph, is defined by an ordered pair G = [V, E], where V is a finite, nonempty set and E is a finite (possibly empty) set of two-tupples representing a relation on V. Members of the set V are called vertices, and members of the set E are called edges. An edge is described by the pair [v(i), v(j)] which indicates that the edge is directed from vertex v(i) to vertex v(j). If the edge [v(i), v(j)] exists, we say that v(j) is adjacent to v(i). A directed graph often is drawn with the vertices as points and the edges as lines like that shown in Graph 1. The arrowhead is directed toward the vertex v(j) of the pair [v(i), v(j)]. In this definition of a graph, we do not include multigraphs (graphs with more than one edge [v(i), v(j)]), nor do we include a graph with edges [v(i), v(i)].

An undirected graph is a special case of the directed graph in which, for every edge [v(i), v(j)], there exists an edge [v(j), v(i)]. Since the directed graph is the more general, we will be concerned only with digraphs.

A graph can be completely specified by an adjacency matrix A. If there are N vertices, then the adjacency matrix is a square matrix with N rows and N columns, such that: $a(i, j) = 1$ if $[v(i), v(j)] \in E$; and
$a(i, j) = 0$ if $[v(i), v(j)] \notin E$.

Table 1 shows the adjacency matrix for Graph 1.

Given a pair of graphs G1 and G2, an isomorphism is a one-to-one mapping $\emptyset$ of the vertices of G1 onto the vertices of G2: $\emptyset(v1(i)) = v2(s)$ and $\emptyset(v1(j)) = v2(t)$, such that v2(s) and v2(t) are adjacent if and only if v1(i) and v1(j) are adjacent.

EXISTING ALGORITHMS

The problem of testing pairs of graphs for isomorphism is known to have an exhaustive search solution of order N!, where N is the number of vertices. Although a number of matrix algebraic techniques have been tried [8, 12], there is no mathematical function which is known to characterize graphs up to isomorphism. Consequently, the problem has been attacked using heuristics [1, 3, 4, 10, 11, 13].

There are a number of conditions which are known to be necessary but not sufficient for isomorphism. One such condition is that for $\emptyset(v1(i)) = v2(s)$ to be an isomorphism, v1(i) and v2(s) must exhibit the same degree. Most published heuristic algorithms use this condition to obtain an initial partitioning of the vertices, which limits some of the required searching. They then employ other conditions as heuristics to further limit the searching. In general, these heuristics represent attempts to characterize a vertex by its relationship with ever increasing numbers of vertices more distantly connected. What seems to be needed is a way of evaluating the relationship of a vertex with all other vertices; i.e., its relationship with the rest of the graph.

THE DISTANCE MATRIX

The distance matrix is a characterization of a graph which offers just this type of information. The distance matrix is an N × N matrix in which the element d(i, j) represents the length of the shortest path between the vertices v(i) and v(j). For every pair of vertices v(i) and v(j), there is a unique distance. If no path exists between the two vertices, the length is defined to be infinite. The elements of the distance matrix are:

d(i, j) = length of the shortest path between v(i) and v(j);

d(i, i) = 0.

The distance matrix for Graph 1 is shown in Table 2. Hakimi and Yau [7] have shown that, for weighted graphs, if G is an N-vertex realization of D without redundant edges, then G is unique. For unweighted graphs, no edge can be redundant.

Given a graph, generation of the distance matrix is a matter of finding the shortest distance between every pair of vertices. A number of algorithms have been developed to construct the distance matrix, many of which are summarized by Dreyfus [5]. Floyd's algorithm [6] provides a means of determining the distance matrix in $O(N^3)$.

AN IMPROVED INITIAL PARTITIONING

Since the distance matrix is a unique representation of a graph, and since it contains information concerning the relationship of vertices in the graph, it offers a means of finding an initial partition which is finer than that obtained by using the degrees of the vertices. Define a row characteristic matrix XR to be an N × (N-1) matrix in which the element xr(i, m) is the number of vertices which are a distance m away from vertex v(i). The row characteristic matrix for Graph 1 is shown in Table 3. Each row of the matrix characterizes the respective vertex. To illustrate, define a class vector C such that the element c(i) represents the class of vertex $v(i)$. Vertices which exhibit identical rows of the characteristic matrix will be assigned the same class. All vertices with the same class are then partitioned into the same cell. For Graph 1, the class vector is $C = [1, 2, 3, 2, 3, 2]$, which means that vertices $v(2)$, $v(4)$, and $v(6)$ are in the same cell of the partition with $v(3)$ and $v(5)$ in another cell.

The characteristic matrix in Table 3 was generated from the rows of the distance matrix. Thus, each element $xr(i, m)$ represents the number of vertices which are of distance m from vertex $v(i)$. Since the distance matrix defines directed graphs, a similar characteristic matrix may be generated from the columns of the distance matrix. Designate the characteristic matrix derived from the columns as XC as shown in Table 4. A composite characteristic matrix X may be defined as the intersection of XR and XC. For illustrative purposes, define the intersection so that $x(i, j) = xr(i, j)$ concatenated with $xc(i, j)$. The composite characteristic matrix for Graph 1 is shown in Table 5. The same procedure, applied to Graph 2 yields the characteristic matrix shown in Table 6.

Algorithm 1 is used to find an initial partitioning for a pair of graphs. Let Graph 1 be G1 and Graph 2 be G2. The resulting class vectors are: $C1 = [1, 2, 3, 4, 3, 2]$, and $C2 = [2, 3, 4, 3, 2, 1]$; which produces the initial partitioning [(1) (2,6) (3,5) (4)] and [(6) (1,5) (2,4) (3)] for G1 and G2 respectively. Note that the initial partitioning using the degree sequence is [(1) (2,3,5,6) (4)] and [(6) (1,2,4,5) (3)]. The initial partition based on the distance matrix is more refined. It will never be less so.

The initial partition typically reduces the amount of work that a heuristic must perform because it tends to limit the size of the remaining search tree. In some cases, it will provide a means of determining immediately that no isomorphism exists for a given pair of graphs. If we define a class count vector K such that the element $k(i)$ is the number of vertices in class i, then $\emptyset(G1) = G2$ is an isomorphism only if $k1(i) = k2(i), \forall i$. After obtaining the initial partition, it is possible to reject pairs of graphs which do not have identical class count vectors.

THE HEURISTIC ALGORITHM

While the initial partitioning generally reduces the size of the search tree, it does not uniquely characterize the vertices of a graph. The second phase of the algorithm consists of selecting possible vertex mappings between the two graphs and checking these mappings for consistency. A vertex mapping $\emptyset(v1(i)) = v2(s)$ is consistent if: (1) every element $d1(i, j) = d2(s, r), \forall j,r$ such that $\emptyset(v1(j)) = v2(r)$ has been determined; and

(2) every element d1(i, k) (v1(k) not previously mapped) has a corresponding d2(s, p) (v2(p) not previously mapped) such that c1(k) = c2(p). Thus, a consistent mapping implies that row i and column i of D1 have corresponding elements in row r and column r of D2 at least for all previously mapped vertices; and that the remaining elements of those rows and columns do not preclude further choices. The heuristic algorithm will choose pairs of vertices v1(i) and v2(s) which are in the same class and investigate the consistency of ϕ(v1(i)) = v2(s). Such a mapping is *asserted* by intersecting the class vector for the appropriate graph with the respective rows and columns of the distance matrices. For illustrative purposes, the intersection will be defined as the concatenation of c1(j), d1(i, j), and d1(j, i), $\forall$ j; and c2(r), d2(s, r), and d2(r, s), $\forall$ r. Such an intersection generates a new class vector, which defines a refined partition. If the mapping is consistent, another pair of vertices can be chosen for mapping. If a partition is reached such that there are no consistent mappings, then a mapping between two dissimilar vertices has been attempted and it is necessary to backtrack to try another mapping.

Algorithm 2 is a recursive algorithm which describes this process. Since a new class vector and a new class count vector will be generated at each level of the recursive process, an index will be added to these vectors. Thus C1(t) will refer to the class vector generated at level t. An element of the vector c1(i, t) refers to the class of vertex v1(i) generated at level t. Similarly, an index will be added to the vector notation for the class count vector K.

To continue the example for which the initial partitioning was determined in the last section, Algorithm 2 will be applied with the corresponding step number indicated in parentheses.

(1) t = 1, C1(t) = [1, 2, 3, 4, 3, 2], C2(t) = [2, 3, 4, 3, 2, 1].
(3) Choose v1(1).
(4) Choose v2(6).
(7) Intersect C1(1) with row 1 and column 1 of D1 yielding [100, 212, 322, 421, 312, 211]. Intersect C1(1) with row 6 and column 6 of D2 yielding [211, 312, 421, 322, 212, 100]. Assign classes based on the partitioning: C1 = [1, 2, 3, 4, 5, 6], C2 = [6, 5, 4, 3, 2, 1].
(9) t = 2.

Note that, although the algorithm must continue verifying the consistency of the mappings, each of the vertices has a unique class assignment so that the mapping at each level is determined. Since there was no arbitrary choice made at any level, either the mapping $\emptyset$ = (1-6, 2-5, 3-4, 4-3, 5-2, 6-1) is an isomorphism or no isomorphism exists for the two graphs. In this case, it is easy to see that the mapping is an isomorphism. However, the algorithm must continue through all six levels to verify consistency.

ANALYSIS OF THE ALGORITHM

In step (3) of Algorithm 2, the choice can be constrained to the smallest class containing an unmapped vertex. This heuristic ensures that the smallest number of choices need to be made at that level. If an unmapped vertex is in a unique class, asserting its mapping may refine the partition as in the example above illustrating the algorithm. In general, if there are k classes of sizes N(1), ..., N(k), then the exhaustive search would require [N(1)! * N(2)! * ... * N(k)!] choices. However, if the smallest N(i) is chosen, then refinements realized by asserting a pair of vertices in that class may reduce the size of a larger N(j) and hence reduce the amount of searching required at the next level.

This algorithm has been implemented using FORTRAN on a Xerox Sigma-7 computer. The time required to obtain the distance matrix is of order N^3. For randomly generated graphs, the algorithm has never had to backtrack, and run time for the algorithm is of statistical order N^2. Graphs of 100 vertices have run in 1.5 minutes including overhead time. For some strongly regular graphs (see Bose [2] and Corneil [3]), the algorithm is required to backtrack. However, pairs of strongly regular graphs of order 25 which are not isomorphic (see Paulus [9]) have been tested and run times average less than one minute.

Other authors reporting implementation run times indicate that their algorithms are useful for a limited number of vertices. However, Corneil and Gotlieb [3, 4] have reported times as low as 0.033 minutes for a 60 vertex graph, but reach 38.7 minutes for a 60 vertex polygon. The algorithm presented here certainly is competitive with existing algorithms and appears useful for problems of realistic size.

```
(1)  Set c1(i) = 0, (i=1, N).
(2)  Compute the row and column characteristic matrices, XR1 for G1 and XR2
     for G2:
     XR1 = {xr1(i, m), (i=1, N; m=1, N-1):
                       xr1(i, m) = |{d(i, j), (j=1, N):d(i, j) = m}|}
     XR2 = {xr2(s, m), (s=1, N; m=1, N-1):
                       xr2(s, m) = |{d(s, r), (r=1, N):d(s, r) = m}|}
     XC1 = {xc1(j, m), (j=1, N; m=1, N-1):
                       xc1(j, m) = |{d(i, j), (i=1, N):d(i, j) = m}|}
     XC2 = {xc2(r, m), (r=1, N; m=1, N-1):
                       xc2(r, m) = |{d(s, r), (r=1, N):d(r, s) = m}|}.
(3)  Intersect XR1 with XC1 and XR2 with XC2.
(4)  CLASS = 0.
(5)  CLASS = CLASS + 1.
(6)  Select some integer k (k=1, N) such that c1(k) = 0.  If none, stop.
(7)  Determine two sets of integers:
     W1 = {i:x1(i, m) = x1(k, m), (m=1, N-1)}
     W2 = {i:x2(i, m) = x2(k, m), (m=1, N-1)}.
(8)  Make class assignments:
     C1(i) = CLASS, {i ε W1}
     C2(i) = CLASS, {i ε W2}.
(9)  Go to 5.
```

ALGORITHM 1: Algorithm to Determine Initial Partitioning.

```
(1)  Set K1(1) = K1 and K2(1) = K2; level t = 1.
(2)  If t = N+1, conclude that the graphs are isomorphic, present the
     mapping, then stop.
(3)  Choose some integer i for which Øv1(i) has not been chosen.
(4)  Choose some v2(s) such that c1(i, t) = c2(s, t) and for which there
     has been no mapping yet chosen for v2(s).
(5)  If no such v2(s) can be found, decrease level to t = t-1.
(6)  If t = 0, conclude that the graphs are not isomorphic and stop.
(7)  Test the consistency of this mapping by intersecting C1(t) with row i
     and column i of D1 to generate C1(t+1).  Similarly intersect C2(t)
     with row s and column s of D2 to derive C2(t+1).
(8)  If K1(t+1) ≠ K2(t+1), go to 4.
(9)  Increase t to t = t+1.
(10) Go to 3.
```

ALGORITHM 2: Algorithm to Test G1 and G2 for Isomorphism.

	1	2	3	4	5	6
1	0	1	0	0	1	1
2	0	0	0	1	1	0
3	0	1	0	1	0	0
4	1	0	1	0	0	0
5	0	0	0	1	0	1
6	1	0	1	0	0	0

TABLE 1: Adjacency Matrix for Graph 1.

	1	2	3	4	5	6
1	0	1	2	2	1	1
2	2	0	2	1	1	2
3	2	1	0	1	2	3
4	1	2	1	0	2	2
5	2	3	2	1	0	1
6	1	2	1	2	2	0

TABLE 2: Distance Matrix for Graph 1.

	1	2	3	4	5
1	3	2	0	0	0
2	2	3	0	0	0
3	2	2	1	0	0
4	2	3	0	0	0
5	2	2	1	0	0
6	2	3	0	0	0

TABLE 3: Row Characteristic Matrix XR for Graph 1.

	1	2	3	4	5
1	2	3	0	0	0
2	2	2	1	0	0
3	2	3	0	0	0
4	3	2	0	0	0
5	2	3	0	0	0
6	2	2	1	0	0

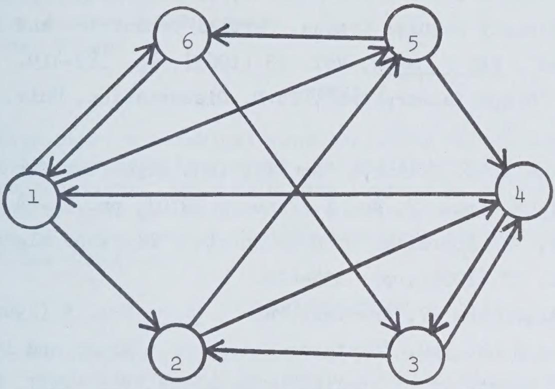
TABLE 4: Column Characteristic Matrix XC for Graph 1.

	1	2	3	4	5
1	32	23	00	00	00
2	22	32	01	00	00
3	22	23	10	00	00
4	23	32	00	00	00
5	22	23	10	00	00
6	22	32	01	00	00

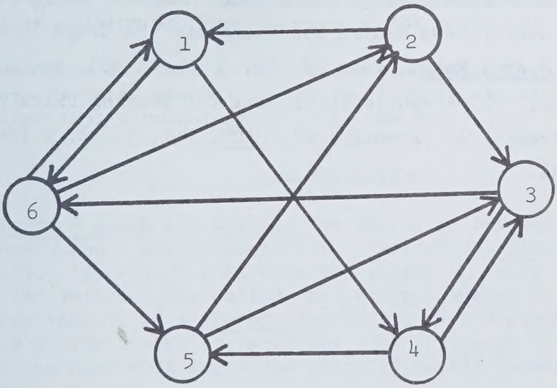
TABLE 5: Characteristic Matrix X for Graph 1.

	1	2	3	4	5
1	22	32	01	00	00
2	22	23	10	00	00
3	23	32	00	00	00
4	22	23	10	00	00
5	22	32	11	00	00
6	32	23	00	00	00

TABLE 6: Characteristic Matrix X for Graph 2.



GRAPH 1.



GRAPH 2.

REFERENCES

1. A. T. Berztiss, "A Backtrack Procedure for Isomorphism of Directed Graphs", JACM, Vol. 20, No. 3 (July 1973), pp. 365-377.
2. R. D. Bose, "Strongly Regular Graphs, Partial Geometries and Partially Balanced Designs", PAC J. MATH, Vol. 13 (1963), pp. 389-419.
3. D. G. Corneil, "Graph Isomorphism", Ph.D. Dissertation, Univ. of Toronto (1968).
4. D. G. Corneil and C. C. Gotlieb, "An Efficient Algorithm for Graph Isomorphism", JACM, Vol. 17, No. 1 (January 1970), pp. 51-64.
5. Stuart Dreyfus, "An Appraisal of Some Shortest Distance Algorithms", OPNS. RES., Vol. 17 (1969), pp. 395-412.
6. R. W. Floyd, "Algorithm 97, Shortest Path", CACM, Vol. 5 (1962), p.345.
7. S. L. Hakimi and S. S. Yau, "Distance Matrix of a Graph and its Realizability", Quarterly of Applied Mathematics, Vol. XXII, No. 4 (1965), pp. 305-317.
8. Frank Harary, "The Determinant of the Adjacency Matrix of a Graph", SIAM Review, Vol. 4, No. 3 (July 1962), pp. 202-210.
9. A. J. L. Paulus, "Conference Matrices and Graphs of Order 26", T.H. Report 73-WSK-06, Technological Univ. Eindhoven, Netherlands.
10. E. Sussenguth, Jr., "A Graph Theoretic Algorithm for Matching Chemical Structures", J.Chem.Doc., Vol. 5, No. 1 (February 1965), pp. 36-43.
11. Y. J. Shah, G. I. Davida, M. K. McCarthy, "Optimum Features and Graph Isomorphism", IEEE-Trans.Syst., Man, Cybern. (May 1974), pp. 313-319.
12. James Turner, "Generalized Matrix Functions and the Graph Isomorphims Problem", SIAM J.Appl.Math., Vol. 16, No. 3 (May 1968), pp. 520-526.
13. Stephen H. Unger, "GIT-A Heuristic Program for Testing Pairs of Directed Line Graphs for Isomorphism", CACM, Vol. 7, No. 1 (January 1964), pp. 26-34.

ON THE PERIODICITY OF AN ITERATION OF FINITE-VALUED FUNCTIONS

PAWEL KERNTOPF
Computation Center
Polish Academy of Sciences
Warsaw, Poland

ABSTRACT

In this paper a graphical representation of the composition of functions, called a composition graph, is applied to studying iterative compositions of functions. By considering properties of iterations of composition graphs we have generalized the previous results [1-4] on periodicity parameters of iterations of Boolean functions to the case of arbitrary finite-valued functions.

INTRODUCTION

We describe here a graphical representation of the composition of functions, called a composition graph, that is conceptually similar to the usual representation of sequential machines and finite automata by transition graphs, or state graphs. Composition graphs are formally equivalent to transition graphs of complete sequential machines or finite automata. Hence, some graphical properties of both classes of graphs are common. However, the application of composition graphs here will differ from that of transition graphs.

By considering periodicity parameters of iterations of composition graphs we are able to generalize the previous results [1-4] on iterations of Boolean functions to the case of arbitrary finite-valued functions.

The iterations of the type considered in the paper correspond to iterative logic networks, i.e. networks built up from identical modules connected in a regular manner. Thus, it is hoped that composition graphs might be useful in solving problems concerning the design of such networks.

We assume a familiarity with basic concepts of graph theory and, in particular, directed graphs. We begin, however, by making precise the graph-theoretical terminology we shall use here.

PRELIMINARIES

A directed graph (or digraph for short) is defined as an ordered pair (V,E) where V and E are respectively, a set of vertices and a set of edges ($E \subseteq V \times V$). If $\langle v_i, v_j \rangle \in E$, then the vertex v_i is called the initial vertex, and the vertex v_j is called the terminal vertex of the edge $\langle v_i, v_j \rangle$. We also say that v_i is a predecessor of v_j. The digraph is finite if it contains a finite number of vertices. The in-degree (the out-degree) of a vertex is the number of edges for which it is the terminal vertex (the initial vertex).

A path from v_0 to v_k is a sequence of vertices $v_0, v_1, \ldots, v_k$ such that $\langle v_i, v_{i+1} \rangle \in E$ for $0 \leq i \leq k-1$, and no edge appears more than once in the sequence. A cycle is a path in which the terminal vertex v_k coincides

---

This work was done while the author was a Visiting Scholar at the Digital Systems Laboratory, Stanford University, Stanford, California, in a program made possible by the Council on International Exchange of Scholars, Washington, D. C.

with the initial vertex v_0. The *length* of a path (a cycle) is defined as the number k, i.e. the number of edges in the path (the cycle).

A digraph is said to be *weakly connected* if its corresponding undirected graph is connected. Each maximal weakly connected subgraph of a digraph G will be called a *component* of G.

COMPOSITION GRAPHS

The class of graphs of interest here is defined as follows:

Definition 1.

A *composition graph* is a quadruple (V,E,L,γ), where (V,E) is a finite digraph, every vertex of which has in-degree 1; L is some set of labels; $\gamma : V \rightarrow L$ assigns to each vertex v a label $\gamma(v)$.

Note that in the definition of a transition graph for complete automata [5,6] it is required that the out-degree of every vertex is equal to 1. Changing directions of arrows is more "natural" for the present application of composition graphs, but this does not change any relevant graphical properties.

Since every vertex of a composition graph has in-degree 1, it follows that every component of a composition graph contains a single cycle.

Definition 2.

The *period of a component* of a composition graph is the length of its unique cycle. The *period p(G)* of a composition graph is the least common multiple (l.c.m.) of periods of all components of G.

Also, if there exists a path from v_i to v_j in a composition graph, then it is unique, and consequently we can use the notation $[v_i,v_j]$ to denote it. The length of the path $[v_i,v_j]$ will be denoted by $\ln[v_i,v_j]$. A *cycle vertex* of a composition graph is any vertex that lies in a cycle. If G is a composition graph, and v is a vertex of G, let c(v) denote the unique component of G containing v. Then a useful notion of height will be defined as follows.

Definition 3.

If $G = (V,E,L,\gamma)$ is a composition graph and $v_i \in V$, then the *height $h(v_i)$ of v_i* is defined as follows:

$$h(v_i) = \min \{\ln[v_j,v_i] \mid v_j \text{ is a cycle vertex of } c(v_i)\}.$$

The *height h(G) of a composition graph G* is the maximum height of any vertex of G.

Thus, given any composition graph, the height and the period of it can be computed by inspection.

Definition 4.

The *iteration of a composition graph* $G = (V,E,L,\gamma)$ is defined as follows:

$$G^1 = G$$

$$G^{i+1} = (V,E,L,\gamma^{i+1}) \qquad \text{for } i \geq 1,$$

where $\gamma^1 = \gamma$, $\gamma^{i+1}(v) = \gamma^i(v')$, and v' is the unique predecessor of v.

Thus G^{i+1} is obtained from G^i by moving all labels a distance 1 along all edges in the direction of the arrows.

A finite family $\{S_1,S_2,\ldots,S_m\}$ of infinite sequences of elements, where $S_j = \{t_i^j\}$ for $1 \leq j \leq m$, is *ultimately periodic* (u.p.), if there exist integers i_0 and i_1 such that

$$t^j_{i_1+i} = t^j_i$$

for every integer $i \geq i_0$ and every $1 \leq j \leq m$. (Also we shall simply say that an infinite sequence S is u.p. if $\{S\}$ is u.p.). The integers i_0 and i_1 will be called an initial length and a period of the family (the sequence), respectively. It is easy to show that the following lemma holds.

Lemma.

If infinite sequences $S_1,\ldots,S_m$ are u.p. and i^j_0, i^j_1 are respectively, an initial length and a period of the sequence S_j for $1 \leq j \leq m$, then the finite family of sequences $\{S_1,\ldots,S_m\}$ is also u.p., the integers $\max\{i^j_0 | 1 \leq j \leq m\}$ and $\text{l.c.m.}\{i^j_1 | 1 \leq j \leq m\}$ being an initial length and a period of the family respectively.

A sequence of composition graphs $G^1, G^2,\ldots$ is obviously u.p., as the number of functions $\gamma : V \to L$ is finite. The following theorem will be fundamental for our further considerations.

Theorem 1.

Given a digraph (V,E), then for every set L and every function $\gamma: V \to L$

$$G^{h(G)+p(G)+1} = G^{h(G)+1},$$

where G is a composition graph (V,E,L,γ).

Proof:

Let c be a component of G, and v be a vertex in c. We shall consider the following cases.

(i) v is a cycle vertex (i.e. $h(v) = 0$)

Then $\gamma^{p(c)+i}(v) = \gamma^i(v)$ for all $i \geq 1$.

So $\gamma^{h(c)+p(c)+1}(v) = \gamma^{h(c)+1}(v)$.

(ii) $h(v) > 0$

Then $\gamma^{h(v)+1}(v) = \gamma(v')$ for some cycle vertex v'

and $\gamma^{h(v)+p(c)+1}(v) = \gamma^{p(c)+1}(v')$. But for a cycle vertex v' we have

$$\gamma^{p(c)+1}(v') = \gamma(v').$$

Thus, $\gamma^{h(v)+p(c)+1}(v) = \gamma^{h(v)+1}(v)$.

From the above it follows that for each component c of G and each vertex v of c we have

$$\gamma^{h(c)+p(c)+1}(v) = \gamma^{h(c)+1}(v).$$

So we have proved that the theorem holds for each component of G. Then by Lemma it holds for the graph G itself.

ITERATION OF FINITE-VALUED FUNCTIONS

Consider a function $f(x_1,\ldots,x_n)$ over a finite domain, i.e. $f: A^n \to A$ and the cardinality of A is finite. We shall call f a finite-valued function. Let $x_{(0)}, x_{(1)}$ be a partition of the set $\{x_1,\ldots,x_n\}$ with k variables in $x_{(0)}$, and assume, without loss of generality, that $x_{(0)} = \{x_1,\ldots,x_k\}$.

Definition 5.

Given a function $f(x_1,\ldots,x_n)$, the iteration of f with respect to k variables in $x_{(0)}$ is defined as follows:

$$f^1 = f$$

$$f^{i+1} = f^i(f(x_1,\ldots,x_n),\ldots,f(x_1,\ldots,x_n),x_{k+1},\ldots,x_n) \quad \text{for } i \geq 1.$$

It was shown (see [1-3]) that if $x_{(0)} = \{x_1\}$, then for every Boolean function f

$$f^3 = f.$$

Recently, Thayse [4] proved that

$$f^4 = f^2$$

for every Boolean function f and every partition $x_{(0)}, x_{(1)}$.

In this paper composition graphs will be applied to studying periodic properties of the iteration of finite-valued functions.

Let $(a_1,\ldots,a_n)$ and $(b_1,\ldots,b_n)$ be elements of A^n. With a finite-valued $f : A^n \to A$ we associate a composition graph $G_f = (V,E,L,\gamma)$ as follows:

1. $V = A^n$
2. $\langle (a_1,\ldots,a_n),(b_1,\ldots,b_n) \rangle \in E$

   if and only if

   $$a_j = \begin{cases} f(b_1,\ldots,b_n) & \text{for } 1 \le j \le k \\ b_j & \text{for } k+1 \le j \le n \end{cases}$$
3. The label $\gamma(v)$ assigned to $v = (a_1,\ldots,a_n)$ is $f(a_1,\ldots,a_n)$.

Then if follows from the definitions that G_f^i is the composition graph of the function f^i.

The following theorem is a generalization of results obtained in [1-4].

Theorem 2.

For every finite-valued function $f : A^n \to A$ and every partition $x_{(0)}$, $x_{(1)}$ of its set of variables

$$f^{h+p+1} = f^{h+1},$$

where $h = |A|$ and $p = \text{l.c.m.}\{1,2,\ldots,|A|\}$.

Proof.

By the definition of the composition graph G_f the last n-k coordinates of the initial and the terminal vertices of every edge in G_f coincide. Thus, G_f is partitioned into $|A|^{n-k}$ disjoint parts determined by values of these coordinates, with $|A|^k$ vertices in each part. Moreover, the first k coordinates of the initial vertex of any edge coincide, too. So, in each part only $|A|$ vertices with identical first k coordinates may have out-degree greater than zero. Thus, the height and the period of each part are less than or equal to $|A|$. Then the height of G_f is less than or equal to $|A|$ and the period of G_f is a divisor of l.c.m. $\{1,2,\ldots,|A|\}$. Thus, by Theorem 1 and the correspondence between G_f^i and f^i we obtain the desired results.

Note that when n = 1 or k = n, then G_f contains only one part having at most $|A|$ vertices with out-degree greater than zero. Thus, the value of p is $|A|$ in this case. For $n > 1$ and $k \neq n$ we have at least $|A|$ such parts, so it is easy to construct functions having a period equal to l.c.m.$\{1,2,\ldots,|A|\}$. Therefore, this is the smallest period of the set of iteration sequences for the set of all finite-valued functions over a domain of cardinality equal to $|A|$.

By an analogous method to that used in the proof of Theorem 2 one can show that if $g = (g_1,\ldots,g_k)$ is a sequence of arbitrary functions over the same set of variables as $f : A^n \to A$, $1 \le k \le n$, then the integers $|A|^k$ and l.c.m.$\{1,2,\ldots,|A|^k\}$ are an initial length and a period of the sequence of iterative compositions defined as follows:

$$C_{f,g}^1 = f(x_1,\ldots,x_n)$$

$$C_{f,g}^{i+1} = C_{f,g}^i(g_1(x_1,\ldots,x_n),\ldots,g_k(x_1,\ldots,x_n),x_{k+1},\ldots,x_n) \quad \text{for } i \ge 1.$$

CONCLUSIONS

We close the paper with a remark that composition graphs can be used for studying not only iterative but arbitrary compositions of functions. The restriction that functions have the same set of variables is not essential, too. The compositions of the type considered in the paper correspond to iterative logic networks, i.e. networks built up from identical modules connected in a regular manner. Thus, it is hoped that composition graphs will be useful in solving problems concerning the design of such networks. Results on estimation of the cardinality of the set of compositions over a given set of funtions could also be obtained in such a way. This constitutes an approach to the current study [7,8] on the synthesis capability of functions.

ACKNOWLEDGMENT

I would like to thank Professor John Wakerly and Dr. Roy Ogus of the Digital Systems Laboratory, Stanford University for reading the manuscript and for their helpful comments and suggestions.

REFERENCES

[1] Lyngholm, C. and W. Yourgrau, "A double-iteration property of Boolean functions," Notre Dame Journal of Formal Logic, Vol. 1, 1960, pp. 111-114.

[2] Simovici, D. A., "Superpositional properties of switching functions," Ann. Univ. Jasi, Vol. 15, 1969.

[3] Reischer, C. and D. A. Simovici, "Associative algebraic structures in the set of Boolean functions and some applications in automata," IEEE Trans. on Computers, Vol. C-20, March 1971, pp. 298-303.

[4] Thayse, A., "On some iteration properties of Boolean functions," Philips Res. Repts., Vol. 28, No. 2, April 1973, pp. 107-119.

[5] Yoeli, M. and A. Ginzburg, "On homomorphic images of transition graphs" J. Franklin Inst., Vol. 278, 1964, pp. 291-296.

[6] Meyer, J. F., "Algebraic isomorphism invariants for graphs of automata," in Graph Theory and Computing, Academic Press, New York and London, 1972, pp. 123-152.

[7] Fujita, Y., T. Kitahashi and K. Tanaka, "Homomorphisms of logical functions and their applications," Conference Record of the 1971 Symposium on the Theory and Applications of Multiple-Valued Logic Design, Buffalo, New York, May 27-28, 1971, pp. 36-41.

[8] Fujita, Y., T. Kitahashi and K. Tanaka, "A consideration on compositions of multi-valued logical functions," Electronics and Communications in Japan, Vol. 55-A, No. 5, 1972, pp. 46-51.

DISCRETE SZEGÖ POLYNOMIALS AND AN O(NlogN) ALGORITHM FOR ERROR EVALUATION*

F. CHIN AND K. STEIGLITZ
Electrical Engineering Department
Princeton University
Princeton, New Jersey 08540

ABSTRACT

Szegö's two-term recurrence formula for orthogonal trigonometric polynomials on the unit circle is shown to lead to a well defined recursive algorithm for generating a system of discrete orthogonal polynomials. An explicit formula for the norm of the n-th polynomial is derived. An identity which corresponds in some respects to the Christoffel-Darboux formula is proved by an approach different from Szegö's. An O(NlogN) algorithm is given for calculating the error of approximation with O(N) orthogonal polynomials.

I. INTRODUCTION

Orthogonal polynomials on the real line are widely used in numerical analysis and curve fitting problems [1]. One way to compute them is by the Gram-Schmidt orthogonalization process. A more efficient algorithm, which makes use of a three-term recurrence formula, was suggested by Householder [2] and Stiefel. A. C. R. Newbery [3] and D. B. Hunter [4] have described analogous three-term recurrence formulas for generating orthogonal trigonometric polynomials. These are useful in dealing with periodic or quasi-periodic functions.

Polynomials orthogonal on the unit circle possess properties which are in some respects simpler than those of polynomials orthogonal on a real interval. They satisfy a (complex) two-term recurrence formula instead of a three-term one. In this paper, we show that this two-term recurrence formula leads to a well defined recursive algorithm for generating a system of discrete orthogonal trigonometric polynomials, which we call Discrete Szegö Polynomials.

We also introduce an asymptotically efficient algorithm for the evaluation of the error of approximation of an arbitrary function f. If f is approximated at N points by $\sum_{k=0}^{n} c_k q_k$ in the least square sense, then the error E is given by $E = (\langle f,f\rangle - \sum_{k=0}^{n} |c_k|^2)^{1/2}$. The obvious method of calculating E takes O(nN) time, but when the sample points are equally spaced on the unit circle we can compute E in O(NlogN) time.

II. Discrete Szegö Polynomials

Szegö [5] gives the two-term recurrence formula for the complex polynomials on the unit circle

---

*This work was supported in part by NSF Grant GK-42048 and the U.S. Army Research Office - Durham under Contract DAHC04-69-C0012.

$$p_{n+1}(z) = z\, p_n(z) - \alpha_n p_n{}^*(z) \qquad (1)$$

where $p_n(z)$ is the $(n+1)^{\text{th}}$ polynomial of precise degree n in which the coefficient of z^n is real and positive; and $p_n^*(z) = z^n \bar{p}_n(z^{-1})$, the reciprocal polynomial. $\bar{p}_n(z)$ denotes the polynomial whose coefficients are the complex conjugates of the corresponding coefficients of $p_n(z)$. For example, if $p_n(z) = \sum_{k=0}^{n} c_k z^k$, then $p_n^*(z) = \sum_{k=0}^{n} \overline{c_{n-k}}\, z^k$.

We consider in our paper discrete polynomials; that is, polynomials evaluated at a set of distinct points, $z_k = e^{i\theta_k}$, where $k=0,1,2,\ldots,N-1$. This corresponds to choosing a measure of the finite type in Szegö's work. By setting $p_0(z)=1$, we can use Eq. (1) recursively to generate a system of discrete polynomials which we call "Discrete Szegö Polynomials". It is easy to show by induction that the coefficient of z^{n+1} in $p_{n+1}(z)$ is unity. In what follows we always assume the $p_k(z)$ satisfy Eq. (1), that $p_0(z)=1$, and that z is on the unit circle.

Definition: The inner product of two complex functions $f(z)$ and $g(z)$ is denoted by $\langle f(z),g(z)\rangle = \sum_{k=0}^{N-1} f(z_k)\overline{g(z_k)}$, and $f(z)$ is orthogonal to $g(z)$ if and only if $\langle f(z),g(z)\rangle=0$.

Definition: A system of polynomials $\{p_n(z)\}$ is said to be orthogonal if $\langle p_n(z),p_m(z)\rangle=0$ for all $n\neq m$, and orthonormal if the system has the extra property that $\langle p_n(z),p_n(z)\rangle=1$ for all n.

Lemma 1: (a) $\langle f(z),h(z)g(z)\rangle=\langle f(z)\bar{h}(z^{-1}),g(z)\rangle$

(b) $\langle f(z)h(z),g(z)\rangle=\langle f(z),\bar{h}(z^{-1})g(z)\rangle$

Proof: Follows from the definition. ▌

The above lemma exhibits a useful property of the inner product; for example, it follows trivially that $\langle zf(z),zg(z)\rangle=\langle f(z),g(z)\rangle$ and also that $\langle z^{-1}f(z),c\rangle=\bar{c}\langle f(z),z\rangle$ where c is a complex scalar.

We shall show that Eq. (1) generates a system of orthogonal polynomials $\{p_n(z)\}$ recursively and lemma 2 establishes a basic step for the inductive proof. Intuitively, it states that if a set of orthogonal polynomials is given, an extra polynomial can be added to it under certain conditions while preserving the orthogonality property of the set.

Lemma 2: If $p_k(z)$ is orthogonal to all $p_j(z)$ where $0\leq j\leq k-1$, and $\langle p_{k+1}(z),p_0(z)\rangle=0$ then $\langle p_{k+1}(z),p_j(z)\rangle=0$ for $0\leq j\leq k$.

Proof: Letting j be in the range $0\leq j\leq k$ and noting that $p_{k+1}=z\, p_k(z)-\alpha_k p_k^*(z)$, we can write $\langle p_{k+1}(z),p_j(z)\rangle= \langle zp_k(z),p_j(z)\rangle-\alpha_k\langle p_k^*(z),p_j(z)\rangle$.

Let $p_{j-1}(z) = \sum_{e=0}^{j-1} c_e z^e$. Noting that $c_{j-1}=1$, we have

$$<zp_k(z),p_j(z)>=<zp_k(z),zp_{j-1}(z)>-\overline{\alpha_{j-1}}<zp_k(z),p^*_{j-1}(z)>$$
$$= <p_k(z),p_{j-1}(z)>-\bar{\alpha}_{j-1}<zp_k(z),z\sum_{e=1}^{j-1}\overline{c_{j-e-1}}z^{e-1}+1>$$
$$= -\overline{\alpha_{j-1}}[<zp_k(z),z\sum_{e=0}^{j-2}\overline{c_e'}p_e(z)>+<zp_k(z),1>]=-\overline{\alpha_{j-1}}<zp_k(z),1>$$

By the same argument, we see that

$$<p^*_k(z),p_j(z)>=-\overline{\alpha_{j-1}}<p^*_k(z),1>$$

Grouping the above three equations together, we obtain

$$<p_{k+1}(z),p_j(z)>=-\overline{\alpha_{j-1}}[<zp_k(z),1>-\alpha_k<p^*_k(z),1>]$$
$$=-\overline{\alpha_{j-1}}<p_{k+1}(z),1> = 0 \qquad ∎$$

Up to now, it has not been shown that a $p_{k+1}(z)$ can be found orthogonal to $p_0(z)$. But lemma 3 will guarantee that α_k is well defined and $<p_{k+1}(z),p_0(z)>=0$ is always possible for all cases with $k<N$.

Lemma 3: If $p_k(z)$ is orthogonal to all $p_j(z)$ where $0 \le j \le k-1$, then

(a) $<p^*_k(z),p_0(z)> = <p_k(z),p_k(z)> > 0$, $k<N$

(b) $<p_{k+1}(z),p_0(z)> = 0$ iff $\alpha_k=<zp_k(z),p_0(z)>/<p^*_k(z),p_0(z)>$

Proof: (a) $<p^*_k(z),p_0(z)> = <z^k\bar{p}_k(z^{-1}),1>$

$$= <z^k, p_k(z)> \quad \text{(by lemma 1)}$$

Since z^k can be expressed by a combination of orthogonal polynomials, and since the coefficient of z^k in $p_k(z)$ is unity, we have

$$z^k = p_k(z) - \sum_{e=0}^{k-1} \beta_e p_e(z) \ .$$

Moreover, $p_k(z)$ is orthogonal to all $p_j(z)$ with $0 \le j \le k-1$, so $<\sum_{e=0}^{k-1}\beta_e p_e(z), p_k(z)>=0$.

Therefore

$$<p^*_k(z), p_0(z)>=<p_k(z),p_k(z)>$$

which is always positive for $k<N$.

(b) Since

$$<p_{k+1}(z),p_0(z)>=<zp_k(z),p_0(z)>-\alpha_k<p^*_k(z),p_0(z)>,$$

it follows immediately that

$$<p_{k+1}(z),p_0(z)>=0 \text{ iff } \alpha_k=\frac{<zp_k(z),p_0(z)>}{<p^*_k(z),p_0(z)>} \ . \qquad ∎$$

The following theorem now shows that an orthogonal set can be generated recursively.

Theorem 1: Let $p_0(z) = 1$

$$p_{k+1}(z) = zp_k(z) - \alpha_k p_k^*(z)$$

$$\text{where} \quad \alpha_k = \frac{\langle zp_k(z),1\rangle}{\langle p_k^*(z),1\rangle}, \quad k<N .$$

Then the system of polynomials is orthogonal.

Proof: By induction on n, the number of polynomials in the system. It is trivially true for n=1 and 2. Assume it is true for $n=k\geq 2$; in particular, p_{k-1} is orthogonal to all $p_j(z)$ for $0\leq j\leq k-2$. If α_{k-1} is defined as in the hypothesis, by Lemma 3, $p_k(z)$ is orthogonal to $p_0(z)$. Furthermore we can conclude that $p_k(z)$ is orthogonal to all polynomials in the system by Lemma 2. Note that the induction works as long as α_k is well-defined, which will be the case when $n\leq N$. ∎

The polynomial $p_N(z)$ has a particularly simple form, as shown in the following theorem.

Theorem 2: $p_N(z) = \prod_{j=0}^{N-1} (z-z_j)$.

Proof: Let's define a mapping φ: $\mathcal{P} \to C^N$ by

$$\varphi(h(z)) = (h(z_0), h(z_1), \ldots, h(z_{N-1}))$$

where $\mathcal{P}$ is the space of complex polynomials, and C^N is an N-dimensional complex vector space. It is obvious that the mapping φ is well-defined. Furthermore it preserves the inner product in these two spaces; that is

$$\langle f(z),g(z)\rangle = \sum_{j=0}^{N-1} f(z_j)\overline{g(z_j)} = \varphi(f(z))\cdot\varphi(g(z)).$$

Now, let's consider the first N polynomials $\{p_j(z)\}$. We have shown in theorem 1 that $\{p_j(z)\}_{j=0}^{N-1}$ forms a set of orthogonal polynomials and therefore $\{\varphi(p_j(z))\}_{j=0}^{N-1}$ is a set of orthogonal vectors in C^N; in other words, a basis for C^N. It is easy to see that none of these vectors vanishes, because if $\varphi(p_m(z))$ is a zero vector in C^N, this implies that $p_m(z)$ which is a polynomial of degree m has N roots and this is impossible for $m<N$. $p_N(z)$, which is generated from $p_{N-1}(z)$, is orthogonal to all $p_j(z)$ where $0\leq j\leq N-1$ and this implies $\varphi(p_N(z))$ is orthogonal to all $\varphi(p_j(z))$ in C^N. This can be true if and only if $\varphi(p_N(z))$ is the zero vector in C^N; i.e. $p_N(z)$ vanishes at all z_e where $0\leq e\leq N-1$. Since $p_N(z)$ is

a monic polynomial of degree N and has N roots at $\{z_e\}_{e=0}^{N-1}$, $p_N(z)$ is forced to be $\prod_{j=0}^{N-1}(z-z_j)$. ▮

Corollary 1 If the inner product is defined on N distinct points on the unit circle, at most N+1 orthogonal polynomials can be generated recursively.

Proof:

$$p_{N+1}(z)=zp_N(z)-\alpha_N p_N^*(z)$$

with
$$\alpha_N = \frac{\langle zp_N(z),1\rangle}{\langle p_N(z),p_N(z)\rangle}$$

But α_N is undefined because by the above theorem both $\langle p_N(z),p_N(z)\rangle$ and $\langle zp_N(z),1\rangle$ vanish. ▮

The vanishing of $p_N(z)$ at the N sample points is a convenient numerical check for a computer program which generates polynomials by this method.

In most cases, it is more convenient to deal with a system of orthonormal polynomials. The following lemma gives an explicit formula for the norm of the polynomials and this simplifies the normalization process.

Lemma 4: $\langle p_{k+1}(z),p_{k+1}(z)\rangle=(1-\alpha_k\bar{\alpha}_k)\langle p_k(z),p_k(z)\rangle$

Proof: Substitute $p_{k+1}(z)$ in terms of $p_k(z)$, $p_k^*(z)$ and α_k by Eq. (1), yielding

$$\begin{aligned}\langle p_{k+1}(z),p_{k+1}(z)\rangle &= \langle zp_k(z),zp_k(z)\rangle+\alpha_k\bar{\alpha}_k\langle p_k^*(z),p_k^*(z)\rangle \\ &\quad -\bar{\alpha}_k\langle zp_k(z),p_k^*(z)\rangle-\alpha_k\langle p_k^*(z),zp_k(z)\rangle\end{aligned}$$

we now have

$$\langle zp_k(z),zp_k(z)\rangle=\langle p_k(z),p_k(z)\rangle \text{ (by lemma 1)},$$

$$\langle p_k^*(z),p_k^*(z)\rangle = \langle z^k\bar{p}_k(z^{-1}),z^k\bar{p}_k(z^{-1})\rangle=\langle p_k(z),p_k(z)\rangle \text{ (by lemma 1)}$$

and if $p_k(z)=z^k+\sum_{e=0}^{k-1} c_e z^e$

$$\begin{aligned}\langle zp_k(z),p_k^*(z)\rangle &= \langle p_k(z),z^{k-1}\,\bar{p}_k(z^{-1})\rangle \\ &= \langle p_k(z),\ \frac{1}{z}+\sum_{e=0}^{k-1}\overline{c_{k-1-e}}\,z^e\rangle \\ &= \langle p_k(z),z^{-1}\rangle=\langle zp_k(z),1\rangle.\end{aligned}$$

Similarly

$$\langle p_k^*(z),zp_k(z)\rangle = \langle z^{-1}\bar{p}_k(z^{-1}),1\rangle$$

From the above identities, we obtain

$$<p_{k+1}(z),p_{k+1}(z)>=$$

$$<p_k(z),p_k(z)>+\alpha_k\bar{\alpha}_k<p_k(z),p_k(z)>-\bar{\alpha}_k<zp_k(z),1>-\alpha_k<z^{-1}\overline{p_k}(z^{-1}),1>$$

$$=<p_k(z),p_k(z)>\left[1+\alpha_k\bar{\alpha}_k-\bar{\alpha}_k\frac{<zp_k(z),1>}{<p_k(z),p_k(z)>}-\right.$$

$$\left.\alpha_k\frac{<z^{-1}\bar{p}_k(z^{-1}),1>}{<p_k(z),p_k(z)>}\right]$$

By applying lemma 3, we have

$$<p_{k+1}(z),p_{k+1}(z)>=<p_k(z),p_k(z)>[1+\alpha_k\bar{\alpha}_k-\bar{\alpha}_k\alpha_k-\alpha_k\bar{\alpha}_k]$$

$$=<p_k(z),p_k(z)>[1-\alpha_k\bar{\alpha}_k]\ .\qquad\blacksquare$$

We shall now demonstrate another important property of discrete Szegö polynomials, an identity which is given by Szegö and which corresponds in some respects to the Christoffel-Darboux formula. We note that our proof is different from Szegö's, and applies only to the discrete case.

Theorem 3: Let $q_k(z)$, $k=0,1,\ldots,N-1$, be a set of orthonormal polynomials obtained by normalizing the $p_k(z)$'s. Then

(a) If $a\neq z$

$$\sum_{j=0}^{n}\overline{q_j(a)}q_j(z)=\frac{\overline{q^*_{n+1}(a)}q^*_{n+1}(z)-\overline{q_{n+1}(a)}q_{n+1}(z)}{1-\bar{a}z}$$

(b) If $a=z$

$$\sum_{j=0}^{n}\overline{q_j(a)}\,q_j(a)=a\overline{q_{n+1}(a)}q'_{n+1}(a)+\bar{a}\,q_{n+1}(a)\overline{q'_{n+1}(a)}-$$

$$(n+1)\overline{q_{n+1}(a)}q_{n+1}(a)$$

Proof: (a) Let $c_m^2=<p_m(z),p_m(z)>$, the normalization factor; then $q_m(z)=\frac{p_m(z)}{c_m}$, $m=0,1,\ldots,N-1$,

we shall prove this identity by induction again. Assume $a\neq z$. For $n=0$,

$$\sum_{j=0}^{n}\overline{q_j(a)}\,q_j(z)=\overline{q_0(a)}\,q_0(z)=\frac{1}{c_0^2}$$

As $q_1(z) = \frac{1}{c_1}(z-\alpha_0)$

$$q_1^*(z) = \frac{1}{c_1}(\bar{\alpha}_0 z-1)$$

So $\dfrac{\overline{q_1^*(a)}q_1^*(z) - \overline{q_1(a)}q_1(z)}{1-\bar{a}z}$

$$= \frac{1}{c_1^2(1-\bar{a}z)}[(\alpha_0\bar{a}-1)(\bar{\alpha}_0 z-1)-(\bar{a}-\bar{\alpha}_0)(z-\alpha_0)]$$

$$= \frac{1}{c_1^2(1-\bar{a}z)}[\alpha_0\bar{\alpha}_0\bar{a}z+1-\bar{a}z-\alpha_0\bar{\alpha}_0]$$

$$= \frac{1}{c_1^2}(1-\alpha_0\bar{\alpha}_0) = \frac{1}{c_0^2} \qquad \text{(by lemma 4)}$$

Assume it is true for $n=k\geq 0$, then

$$\sum_{j=0}^{k+1} \overline{q_j(a)}\, q_j(z) = \frac{\overline{q_{k+1}^*(a)}q_{k+1}^*(z)-\overline{q_{k+1}(a)}q_{k+1}(z)}{1-\bar{a}z} + \overline{q_{k+1}(a)}q_{k+1}(z)$$

$$= \frac{\overline{q_{k+1}^*(a)}\; q_{k+1}^*(z)-\bar{a}z\;\overline{q_{k+1}(a)}q_{k+1}(z)}{1-\bar{a}z}$$

Using $q_{k+2}(z) = \frac{1}{c_{k+2}}[z\, p_{k+1}(z)-\alpha_{k+1}p_{k+1}^*(z)]$, and the corresponding expression for $q_{k+2}^*(z)$, we can show that

$$\overline{q_{k+2}^*(a)}q_{k+2}^*(z) - \overline{q_{k+2}(a)}q_{k+2}(z)$$

$$= \overline{q_{k+1}^*(a)}\; q_{k+1}^*(z) - \bar{a}z\;\overline{q_{k+1}(a)}q_{k+1}(z)$$

(b) When a=z, formula (b) can be obtained by using L'hopital's rule or by mathematical induction. Since the proof is rather straightforward but long, it is omitted here. ▮

III. Error Evaluation

In many cases, it is commonplace to fit an arbitrary function f(z) by a polynomial in the least square sense. Let the approximating polynomial be $\sum_{j=0}^{n} c_j q_j(z)$. If the values of f(z) at N points on the unit circle are given, then the error of approximation can be written as

$$E = (\langle f(z), f(z)\rangle - \sum_{j=0}^{n} |c_j|^2)^{1/2}$$

$$= (\sum_{j=0}^{N-1} f(z_j)\overline{f(z_j)} - \sum_{j=0}^{n} |c_j|^2)^{1/2} \qquad (2)$$

Since $\{q_j\}_{j=0}^{n}$ is a set of orthonormal polynomials,

$$c_j = \langle f(z), q_j(z)\rangle = \sum_{k=0}^{N-1} f(z_k)\overline{q_j(z_k)}$$

The evaluation of each c_j takes $0(N)$ time, and so the error E will take $0(nN)$ time by the obvious approach. Now we introduce an efficient algorithm for the computation of E which takes $0(N\log N)$ time when the sample points are distinct and equally spaced. If n is greater than $0(\log N)$, then the method is asymptotically more efficient.

Let $z = e^{i(\omega_0+k\omega)}$, $k=0,1,\ldots,N-1$. So

$$\sum_{j=0}^{n} |c_j|^2 = \sum_{k=0}^{N-1} \sum_{e=0}^{N-1} f(z_k)\overline{f(z_e)}(\sum_{j=0}^{n} \overline{q_j(z_k)}q_j(z_e))$$

$$= \sum_{\substack{k=0 \\ k\neq e}}^{N-1} \sum_{e=0}^{N-1} f(z_k)\overline{f(z_e)} \left[\frac{\overline{q^*_{n+1}(z_k)}q^*_{n+1}(z_e) - \overline{q_{n+1}(z_k)}q_{n+1}(z_e)}{1 - \bar{z}_k z_e}\right]$$

$$+ \sum_{e=0}^{N-1} f(z_e)\overline{f(z_e)}(\sum_{j=0}^{n} \overline{q_j(z_e)}\, q_j(z_e))$$

Let

$$f'(z_e) = \begin{cases} f(z_e)\overline{q^*_{n+1}(z_e)} & 0 \leq e \leq N-1 \\ 0 & N \leq e \leq 2N-1 \end{cases}$$

$$f''(z_e) = \begin{cases} f(z_e)\overline{q_{n+1}(z_e)} & 0 \leq e \leq N-1 \\ 0 & N \leq e \leq 2N-1 \end{cases}$$

$$g(z_k) = \begin{cases} (1-e^{ik\omega})^{-1} & k\neq 0 \text{ and } -N+1 \leq k \leq N-1 \\ 0 & k=0 \end{cases}$$

and

$$g'(z_k) = \begin{cases} g(z_k) & 0 \leq k \leq N-1 \\ 0 & k=N \\ g(z_{k-2N}) & N < k \leq 2N-1 \end{cases}$$

Then we have

$$\sum_{j=0}^{n} |c_j|^2 - \sum_{e=0}^{N-1} \overline{f(z_e)} f(z_e) \Big(\sum_{j=0}^{n} \overline{q_j(z_e)} q_j(z_e)\Big)$$

$$= \sum_{e=0}^{2N-1} \overline{f'(z_e)} \sum_{k=0}^{2N-1} f'(z_k) g_{e-k} - \sum_{e=0}^{2N-1} \overline{f''(z_e)} \sum_{k=0}^{2N-1} f''(z_k) g_{e-k}$$

$$= \sum_{e=0}^{2N-1} \overline{F'_e} F'_e G'_e - \sum_{e=0}^{2N-1} \overline{F''_e} F''_e G'_e$$

$$= \sum_{e=0}^{2N-1} G'_e (|F'_e|^2 - |F''_e|^2) \qquad (3)$$

where F'_e, F''_e and G'_e are the Discrete Fourier Transforms of f', f" and g' respectively. The above form allows us to evaluate the error of approximation in $0(N\log N)$ time, assuming certain functions are precomputed beforehand for the frequency ω of interest.

Theorem 4: Let $z_k = e^{j(\omega_0 + k\omega)}$ and assume that $q_{n+1}(z_k)$, k=0,..., N-1, and $\sum_{j=0}^{n} \overline{q_j(z_k)} q_j(z_k)$, k=0,...,N-1 are precomputed. Then E can be computed in $0(N\log N)$ time.

Proof: The algorithm for computing E is implicit in the above derivation and has the following steps:

Step 1: The computation of <f,f> takes 0(N) time

Step 2: The definition of f',f" and g' takes 0(N) time

Step 3: The D.F.T. of f',f" and g', i.e. F',F" and G' respectively, can be done by F.F.T. in $0(N\log N)$ time.

Step 4: The calculation of $\sum_{e=0}^{N-1} f(z_e)\overline{f(z_e)} \Big(\sum_{j=0}^{n} \overline{q_j(z_e)} q_j(z_e)\Big)$ takes 0(N) time.

Note: The requirement that $\sum_{j=0}^{n} \overline{q_j(z_k)} q_j(z_k)$, k=0,...,N-1 are precomputed introduces no special restriction to the theorem since they can be calculated at the same time as the set of orthogonal polynomials.

Step 5: $\sum_{e=0}^{2N-1} G'_e (|F'_e|^2 - |F''_e|^2)$ takes 0(N) time.

Step 6: The evaluation of $\sum_{j=0}^{n} |c_j|^2$ by Eq. (3) takes constant time.

Step 7: The computation of E by Eq. (2) also takes constant time.

Notice that the denominator $(1-\overline{z_k}z_e)$, k≠e, never vanishes, since we have assumed the z_i are distinct.

Further simplification of the recursive formulas can be achieved by taking the points z_k in conjugate pairs on the unit circle. The two-term recurrence formula for generating orthogonal polynomials can also be modified and generalized for points on a circle with radius other than unity. Further results and applications will be reported in a future paper.

References

1. G.E. Forsythe, "Generation and Use of Orthogonal Polynomials for data fitting with a Digital Computer," J. SIAM, Vol. 5, 1957; pp. 74-88.

2. A.S. Householder, Principles of Numerical Analysis, McGraw-Hill, New York, 1953.

3. A.C.R. Newbery, "Trigonometric Interpolation and Curve-Fitting," Math. of Comp., Vol. 24, 1970; pp. 869-876.

4. D.B. Hunter, "Algorithm 320: Harmonic Analysis for Symmetrically Distributed Data," CACM, Vol. 11, 1968; pp. 114-5.

5. G. Szegö, Orthogonal Polynomials, Amer. Math. Soc. Colloquium Publications 23, rev. ed. (1959).

ON THE COMPLEXITY OF A CLASS OF CONCENTRATION SWITCHING NETWORKS

GERALD M. MASSON
Department of Electrical Engineering
University of Pittsburgh
Pittsburgh, Pennsylvania 15261

ABSTRACT

This paper developes a class of concentration switching networks capable of providing one input-to-one output types of interconnections between a set of input and output terminals where all input terminals are identical. The main results concern the complexity of the networks in terms of the number of switches required for the attainment of some particular network performance, and in terms of the number operations necessary to affect some particular network action.

Introduction

This paper considers the complexity of a class of switching networks in terms of the number of switches necessary for the attainment of a certain network performance and in terms of the number of operations necessary to affect some particular network action. A *switching network* in general can be described as a set of interconnected single-pole, single-throw switches designed to connect together network inputs with network outputs in various combinations. This is schematically shown in Figure 1. A *concentration switching network* has the particular properties that all connections to be made through it are of a one-input to one-output nature, and that all inputs are identical. Thus, a *request* for a connection through a concentration switching network is simply a request that a particular idle output be connected to any idle input; a request is said to be *satisfied* when a sequence of switches can be found which when closed provides a *path* between this particular idle output and some idle input. Clearly, this path must not intersect any other existing path in the network. The set of closed paths existing through the network at any one time is usually called the *state* of the network at that time.

Concentration switching networks are playing an increasingly important role in many communication switching systems and modular computer designs. For example, consider a set of N processing devices, one connected to each input of the concentration switching network; and another set of M I/O devices, one connected to each output of the concentration switching network, each of which, at

---

This work was supported by National Science Foundation Grant GK-42077

various times, can request a connection to a processor. In such a situation, the concentration switching network used to provide the proper connections of processing devices and I/O devices strongly influences the operation of the entire system.

Aspects of Network Performance

In order that the results of this paper can be developed here, some switching network terminology must be explained.

The capacity, C, of a concentration switching network is the maximum number of simultaneous connections which the network is designed to handle at any one time. When C is less than the number of inputs, the network is called a partial concentration switching network. Clearly, C can never be greater than the number of inputs.

A concentration switching network is said to be incrementally rearrangeable with capacity C if for any state of the network in which at most C-1 connections exist from the inputs to the outputs, any new request requiring that some idle output be connected to any idle input can be satisfied, perhaps at the expense of rearranging some of the existing paths through the network.

The distance of an incrementally rearrangeable network is the maximum number of paths which at any time must be rearranged in order that a new request can be satisfied. Thus, the distance can be considered to be a quantitative measure of the number of operations necessary to take a network from a given state to some new state which satisfies all the connections of the old state as well as the new request. A network of distance zero is said to be nonblocking.

A concentration switching network is said to be graded if the switches of the network are grouped into consecutive ranks. Graded networks have the useful property that each of the switches is not dedicated to a specific connection. Instead, each switch can potentially be used in many connections. This is important in terms of incremental rearrangeability and distance.

Principal Results

Clearly, given N inputs and M outputs, $N < M$, a concentration switching network with CM switches can be constructed simply by providing a switch between each output and any C inputs. Such a network would have distance zero, and, accordingly, CM switches can be considered to be a benchmark against which all other concentration switching networks must be compared.

In order to consider alternative constructions for concentration networks, the paper uses a graph-theoretic model of concentration switching networks similar to models developed to study other types of switching networks [1-4]. The model will not be elaborated upon here except to say that the edges represent switches, and the vertices represent inputs, outputs, or internal terminals of the network. For example, if $N = 4$, $M = 6$, and $C = 4$, the distance zero type of network described in the previous paragraph would be modeled as a bipartite graph with 24 switches as shown in Figure 2.

The notion of a set of distinct representatives (SDR) will be useful in the following. Given a set of subsets $(O_1, \ldots, O_M)$, where $O_j \subseteq I = (1, \ldots, N)$, then an SDR of $(O_1, \ldots, O_M)$ is a set of elements $(i_1, \ldots i_M)$ where $i_j \in O_j$ and $i_j \neq i_k$ if $j \neq k$.

The main results of this paper consist of a constructive development using Hall's Theorem concerning SDR's [5] for a class of graded, incrementally rearrangeable concentration switching networks.

Hall's Theorem: The above sets $(O_1, \ldots O_M)$ have an SDR of and only if the union of any t of the O_j's, $1 \leq t \leq M$, contains at least t elements.

Now, consider a graded concentration switching network of rank one. Denote the N input terminals by the set $I = (1, \ldots, N)$. Furthermore, associate with each of the M output terminals a set O_j, $j = 1, \ldots, M$, where $O_j - I$ and the elements of O_j correspond to the input terminals to which output terminal j has a possible path via some switch. For example, for the network of Figure 2, $O_i = I$, $i = 1, \ldots 6$. Clearly, to provide C simultaneous paths between C specified output terminals, say $j(1), \ldots, j(C)$ and any C distinct input terminals it is necessary and sufficient to find an SDR for $(O_{j(1)}, \ldots, O_{j(C)})$. Thus, switches must be placed in the network to satisfy this condition.

Suppose, now, that $M \leq C^N_{C-2}$, where C^p_q is the binominal coefficient $\binom{p}{q}$, and suppose further that switches were placed between the input and output terminals such that each output had a possible connection via some switch to C-2 of the inputs, and that no other output had switches to that precise set of C-2 inputs. For example, suppose $N = 4$, $M = 6$, and $C = 4$. Then $C^4_2 = 6$. Figure 3 shows a rank one structure where each of the 6 outputs has a switch between it and a distinct choice of 2 of the 4 inputs. Choose any C of the outputs, say $j(1), \ldots, j(C)$, and assume that paths to $j(1), \ldots, j(C-1)$ have been provided in the network. Suppose that the path to output $j(C)$ cannot be provided. By Hall's Theorem, this implies that $O_{j(1)}, \ldots, O_{j(C)}$ contains at most C-1 elements among them. In other words, the C-2 connections via a switch from each of these C inputs have gone to the same set of C-1 inputs. But, this is impossible since at most $C^{C-1}_{C-2} = C-1$ outputs could have switches to those inputs. Hence, this connection to $j(C)$ can be provided. Furthermore, an almost identical argument leads to the conclusion that at most 2 of the existing connections must be rearranged to provide the path for $j(C)$. Thus, we have a distance 2 network for incremental assignments with $(C-2)\,M$ switches.

Now, suppose that $M > C^N_{C-2}$. Let $K = C^N_{C-2}$, and suppose $M \leq C^K_{C-2}$. Then, a rank two network can be constructed where there are N input terminals, $K = C^{K}_{C-2}$ internal terminals, and M output terminals, and the switches are placed between the output terminals and the internal terminals using the same rule as before. Namely, choose C-2 distinct interval terminals for each output terminal and place a switch between them.

For example, of N = 4, M = 15, and C = 2, Figure 4 represents a graded switching network of rank two which has 42 switches. Since each of the ranks is rearrangeable, clearly, the entire network must be rearrangeable. However, in order to rearrange to provide a new connection, it might now be necessary to rearrange 2 existing connections per rank. Such a network is, therefore, of distance 4.

The extension to graded networks of rank j should now be clear. Such networks would require up to 2j rearrangements to satisfy a new request. Note that the first rank would have N input terminals connected to, say, K_1 internal terminals; the second rank would then have K_1 internal terminals connected to, say, K_2 internal terminals; . . . ; and the j^{th} rank would have, say, K_j internal terminals connected to the M output terminals. Thus, the (C-2)M switches of the j^{th} rank would be significantly larger than the number of switches of the previous rank. Hence, the resulting network can be said to have O((C-2)M) switches.

Conclusions

Some preliminary results have been presented concerning the complexity of a class of rearrangeable concentration switching networks. It was seen that the networks have O((C-2)M) switches, and require up to 2j rearrangements of existing connections to satisfy a new request, where j is the number of ranks in the network.

The significance of the results of this paper is not only in the development of a new class of concentration switching networks, but also in that the results suggest the possibility of many other classes of networks based on this given class. The inherent connecting capability of the structures described seems to be quite robust such that their application to other connecting problems seems worthy of future research.

REFERENCES

1. V. E. Beneš, Mathematical Theory of Connecting Networks and Telephone Traffic, Academic Press, Inc., New York, 1965.

2. D. G. Cantor, "On Non-Blocking Switching Networks," Networks, Vol. 1, pp. 367-377, 1971.

3. N. Pippenger, "The Complexity Theory of Switching Networks," Sc. D. Thesis, Department of Electrical Engineering, M. I. T., 1973; Technical Report 487, Research Laboratory of Electronics, M. I. T., 1973.

4. G. M. Masson and B. W. Jordan, J., "Generalized Multi-Stage Connection Networks," Networks, Vol. 2, pp. 191-209, 1972.

5. L. Mirsky, Transversal Theory, Academic Press, Inc., New York, 1971, Chapter 2.

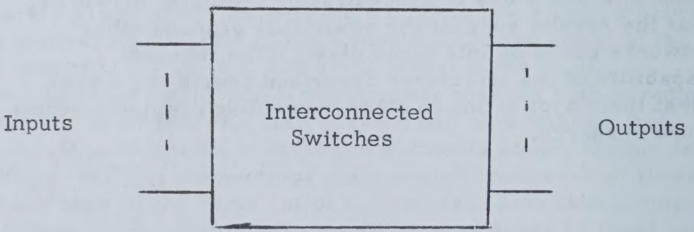


Figure 1. Schematic of a switching network.

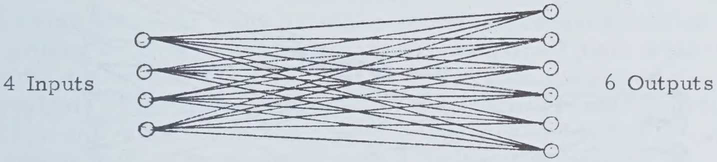

4 Inputs
6 Outputs

Figure 2. A bipartite graph representing a concentration switching network with 4 - inputs and 6 - outputs of capacity 4 having 24 switches.

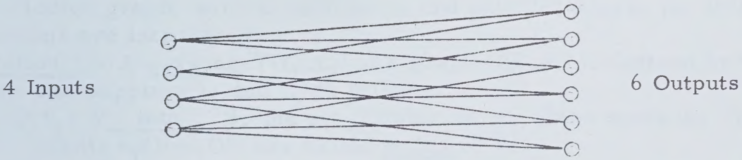

4 Inputs
6 Outputs

Figure 3. Graph representation of a 4 - input, 6 - output concentration switching network of capacity 4 and distance 2 requiring only 12 switches and 1 rank.

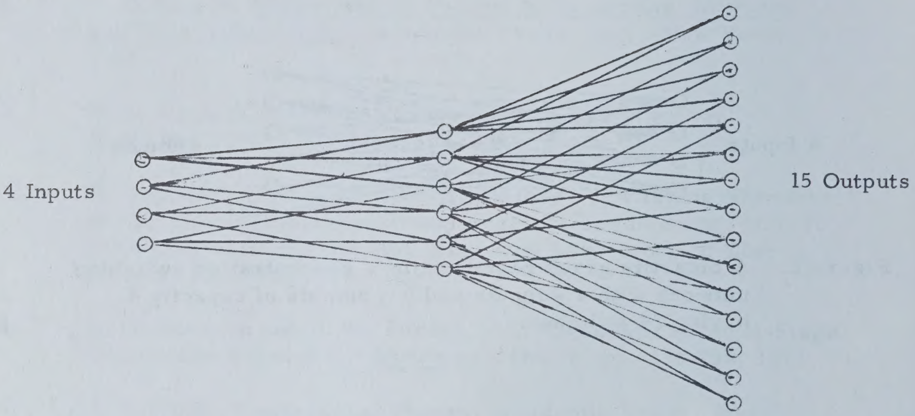

4 Inputs
15 Outputs

Figure 4. Graph representation of a 4 - input, 15 - output concentration switching network of capacity 4 and distance 4 requiring 42 switches and 2 ranks.

GRAPH DRAWING - A THEORETICAL FOUNDATION (')

G. COTTAFAVA, member IEEE and O. D'ANTONA
Istituto di Elettronica, Facoltà di Ingegneria
UNIVERSITA' DI PAVIA
27100 PAVIA, ITALY

ABSTRACT

Graph drawing occurs in many problems related to computer-aided design such as printed or integrated circuit layout, industrial plant layout, traffic network, and so on. The first part of this report is devoted to a theoretical analysis of the problem in view of its practical implementation on a computer. Half-plane drawings are so introduced and it will be shown that all graphs have at least one of such drawing. Condition on a given drawing to be half-plane are given also. The second one introduces the vertex-reachability-graph, that possesses uniqueness property for a wide class of planar graphs. As an application of this concept, a new solution of a drawing of a planar graph with constraints is given.

1. INTRODUCTION

During last years many papers concerning the drawing of a planar graph are appeared. Conversely, few papers [1], [2] treat the graph-drawing problem from a general point of view. Undoubtedly, planar graphs have found many applications by current technology. The uniqueness of representation for a large class of planar graphs [3] is an important property that make easy their drawing. Unfortunately this property cannot extended to the non-planar graphs, for which the problem of a good representation is proposed and his properties presented.

Current definitions on graph theory may be found in [4]. Let G(V,E) be a non-oriented graph, without self-loops and parallel edges, the following definitions are introduced:

Definition 1 - A drawing $T(t_1,t_2)$ of a graph G(V,E) is defined by two one-to-one mappings t_1 and t_2 as follows:

i) t_1: V $\xrightarrow{\text{into}}$ S, the set S being an euclidean surface. The elements $n_i \in N=t_1(V)$ are called nodes of T.

ii) t_2: E $\xrightarrow{\text{into}}$ J_s, J_s being the set of all possible open Jordan arcs of S, whose endpoints are nodes of N. The elements $t_2(v_i,v_j) \in J_E = t_2(E)$, called arcs of T, are subject to:

$$t_2(v_i,v_j) \not\ni n_k \quad , \forall k \neq i,j .$$

A drawing is denoted also by $T(N,J_E)$. The surface S is divided by the drawing into connected regions, in such a way that a Jordan arc connecting the points lying in different regions intersects the drawing edges (or its endpoints) once at least. Obviously a graph has infinite drawings.

(') This work was partially supported by C.I.O.C.-C.N.R., Bologna (Italy).

An interesting sub-class of drawings is the half-plane ones. They represents a valuable tool in solving some problems posed by layout applications. This paper mainly deals with this subclass, defined as follows:

Definition 2 - Let γ be a closed Jordan line of S, a *half-plane drawing* T_γ is a good drawing [2] such as:

$$J_E \cap \gamma = N;$$

that is:

i) all the nodes of T are on the γ-line,

ii) no edge of T crosses the γ-line.

Half-plane drawings have the following properties:

P1. The surface S is partitioned by the γ-line into two regions R_e *(exterior region)* and R_i *(interior region)* [5]. So a partition $\pi = \{E_e, E_i\}$ exists such as $t_2(E_e) \subset R_e$ and $t_2(E_i) \subset R_i$.

P2. A half-plane drawing is defined also by giving:
  i) the ordering of nodes along the γ-line,
  ii) the set E_e (or E_i).

P3. Every graph G(V,E) has at least one half-plane drawing $T_\gamma(t_1, t_2)$. E.g., the trivial drawing defined by $E_e = \emptyset$ (or $E_i = \emptyset$) always exists.

P4. Every half-plane drawing admits a new drawing on a plane Σ, in such a way that:
  i) the closed γ-line of S is transformed into a straight line γ' of Σ, that partitions Σ into two half-planes named H_e and H_i;
  ii) the set E_e (E_i) is drawn on H_e (H_i).

The name given to the half-plane drawing is entirely justified by this last property.

2. HALF-PLANE DRAWING'S PROPERTIES

The computer description (see P2 of par. 1) of a drawing of a graph (planar or not) is made more easy if the drawing itself has half-plane property. The foregoing statement drives us to found a criterion to test if a given drawing is half-plane or not.

Definition 3 - A *vertex-reachability-graph* (V.R.G.) $R(V_R, E_R)$ is a graph corresponding to a drawing $T(N, J_E)$, of G(V,E), constructed in the following way:

i) $v_i \in V_R \Leftrightarrow n_i \in N$.

ii) $(v_i, v_j) \in E_R \Leftrightarrow$ an open Jordan arc $J(n_i, n_j)$, closed by the nodes $n_i = t_1(v_i)$, $n_j = t_1(v_j) \in N$ exists such that it crosses no arc of J_E.

Some interesting properties of V.R.G. are presented in the following lemmas. They allow to construct a γ-line in a half-plane drawing of G.

Lemma 1 - Let $R(V_R, E_R)$ be the V.R.G. of a drawing $T(N, J_E)$, a subgraph $R° \subseteq R$ is complete iff the corresponding subset of nodes $N° \subseteq N$ is on the bound of one region of T.

For the sake of brevity the proof of this lemma is not reported here.

Lemma 2 - Let $G(V,E)$ be a 3-connected graph, $R(V_R,E_R)$ the V.R.G. of a drawing $T(N,J_E)$ of G, and (v_i,v_j) an edge of R. If (v_i,v_j) E, then the corresponding Jordan arc (n_i,n_j) is traceable in one and only one region of T without it crosses arc of J_E.
Proof - By subdivision [1] a planar representation T_p of T is obtained. It is easy to shown [6] that T_p is 3-connected also. From a theorem proved by Bruno et al. [7, Theorem 3], lemma follows immediately.

Theorem 1 - A 3-connected drawing $T(N,J_E)$ is half-plane iff the graph $R(V_R,E_R)$ is hamiltonian.
Proof - The necessity of the condition is proved simply by observing that each arc of the γ-line is an edge of R. The sufficiency is shown by drawing in T each edge of the hamiltonian circuit contained in R without crossings. Let $\tilde{\gamma}$ be this hamiltonian circuit, then from lemma 2 intersects no arc of T. Nevertheless, two situations can arise: a) the $\tilde{\gamma}$-line does not cross itself, b) the $\tilde{\gamma}$-line crosses itself. Clearly, in case a) the theorem is so proved, and $\tilde{\gamma}=\gamma$. In case b) a transformation of $\tilde{\gamma}$-line into a γ-line can be achieved by the following way. Let (n_i,n_j) and (n_k,n_l) be two edges of the $\tilde{\gamma}$-line that cross each-other, $\mathfrak{R}$ the region of T whose bound contains (n_i,n_j,n_k,n_l), and $N°$ the set of nodes of $\mathfrak{R}$. For lemma 1 the subgraph $R°$ on $N°$ is complete. So, in $\mathfrak{R}$ the edges (n_i,n_k), (n_i,n_l), (n_j,n_k), (n_j,n_l) exist. Therefore we can choose two of such edges in such a way that the $\tilde{\gamma}$-line does not cross in T. If more than one crossing exists in $\mathfrak{R}$, it can be shown [8,p. 13] that the crossings are removed by repeating the procedure outlined before.

Let $\mathcal{T}_p(G)$ be the collection of all planar drawings of a graph G, some properties that hold for all the drawings $T\in\mathcal{T}_p(G)$ are now established.

Lemma 3 - Let T be a planar drawing of $G(V,E)$ and $R(V_R,E_R)$ its V.R.G., then the following relationship holds: $E\subseteq E_R$.
Proof - Each edge $(v_i,v_j)\in E$ is a Jordan arc of T without crossings, being T planar. Then, from definition 3 lemma holds.

Let $\mathcal{T}_p(G)$ be the set of all planar drawings of a graph G, the following theorems hold.

Theorem 2 - Let R be a hamiltonian V.R.G. of a 3-connected planar drawing T of G, then all $T\in\mathcal{T}_p(G)$ are half-plane drawings.
Proof - For theorem 1 the drawing T is a half-plane one. Being G a 3-connected graph, it is uniquely embeddable [9, Theorem 11](1) on the plane.

Corollary 1 - Each $T\in\mathcal{T}_p(G)$ has the same V.R.G., iff G is 3-connected.

Theorem 3 - Let G be a 3-connected planar hamiltonian graph, then all $T\in\mathcal{T}_p(G)$ are half-plane drawings.
Proof - From lemma 3 follows that a V.R.G. of a drawing $T\in\mathcal{T}_p(G)$ is 3-connected and hamiltonian. From theorem 2, the statement follows.

Corollary 2 - Let G be a 4-connected planar graph, then all $T\in\mathcal{T}_p(G)$ are

half-plane drawings.
Proof - Being a 4-connected planar graph a hamiltonian one [10], from theorem 3 corollary follows.

The foregoing theorems are suitable to restate that all triangulations are 3-connected [11].
Definition 4 - A triangulation T_n on n vertices is a planar graph, whose regions has three edges as boundary.

Lemma 4 - Let R be the V.R.G. of a triangulation T_n, then $T_n \equiv R$ (2).

Theorem 4 - T_n is 3-connected.
Proof - From lemma 4 and corollary 1 theorem follows.

3. APPLICATION TO THE ENGL-MLYNSKI PROBLEM

The wire routing on a printed circuit board (or IC chip) is subjected to certain constraints which are specified by the system designer. Engl-Mlynski [12] have introduced such constraints, which they have referred to as constraints C1 and C2. In this section, a new approach of the Engl-Mlynski problem by the aid of V.R.G. concept is proposed. Notice that constraint C1 implies constraint C2; for this reason the problem with constraint C2 is treated before.

Theorem 5 - Let $V' \subset V$ be the set of "terminal" (3) vertices of a planar graph G, a drawing $T \in \mathcal{T}_p(G)$ meets the constraint C2 iff the corresponding V.R.G. contains a complete subgraph spanning V'.
Proof - If V.R.G. is complete, then by lemma 1 the corresponding subset N' of T is placed on the bound of one region of T. By proper stereographic projections this region can be mapped onto the exterior region of the plane. If T meets C2, then the subset N' is on the bound of the same region of T and by lemma 1 theorem follows.

Moreover, constraint C1 requires that the set V' is placed in a given sequence on the bound of one region of T whose set of vertices is V" ($V' \subseteq V''$). The set V" is properly described by the following
Definition 5 - A terminal clique K (V", L) of a drawing $T \in \mathcal{T}_p(G)$ is the maximal complete subgraph of the V.R.G. of T containing the subset of terminal vertices V'.
Clearly $V' \subseteq V'' \subseteq V$ (4).

Lemma 5 - The subgraph $Z = K \cap T$ is a circuit.
Proof - Being K a clique of the V.R.G. of T, then for lemma 1 its vertices are on the bound of a region of T, and being T planar, the lemma follows.

Theorem 6 - Let V' V be the set of "terminal" vertices given in a "specified sequence" of a planar graph G, and $T \in \mathcal{T}_p(G)$ be a drawing meeting constraint C2, T meets constraint C1 iff the set of vertices V' is encountered along Z in the specified sequence.

Proof - "Ab absurdo", if the sequence of the set of vertices in Z is not the same as in T, then for the preceeding lemma Z is not a circuit of T. Conversely, if T meets C1, then V' is on the bound of a region in the specified sequence, for lemma 5 theorem follows.

4. FINAL REMARK

In this paper two new concept are introduced: the half-plane drawing and the vertex-reachability graph. The first one has been very suitable to develope an algorithm that allows to treat graph by a computer [8], performing operations as drawing, storing or interactive modifications. The second one appears as a good tool to find some properties of the planar drawings; e.g. the uniqueness of V.R.G. for a class of planar graphs and so on.

---

(1) 3-connected graphs are a proper subset of the class of uniquely embeddable planar graphs as shown by theorem 3.5 in [7]. Theorem 2 and the following ones hold also for this class.

(2) In the lemma we have denoted also by T_n a planar drawing of the graph T_n.

(3) Terminology not defined here is taken from [12].

(4) Notice that $V'' \equiv V$ iff T is outerplanar.

5. REFERENCES

[1] W. T. Tutte - Toward a Theory of Crossing Numbers - Journal of Combinatorial Theory, 8, pp. 45-53, 1970.

[2] R. K. Guy - Latest Results on Crossing Number - Recent Trends in Graph Theory Conf., Springer-Verlag, Berlin 1971, pp. 143-156.

[3] H. Fleischner - The Uniquely Embeddable Planar Graphs - Discrete Mathematics, Vol. 4, pp. 347-358, 1973.

[4] F. Harary - Graph Theory - Addison Wesley, Reading 1969.

[5] G. D. Whyburn - Topological Analysis - Princeton University Press, Princeton 1958.

[6] W. T. Tutte - A Theory of 3-connected graphs - Indag. Math., Vol. 23, pp. 441-455, 1961.

[7] J. Bruno, K. Steiglitz, L. Weinberg - A New Planarity Test Based on 3-Connectivity - IEEE Trans. on Circuit Theory, CT-17, pp. 197-206, 1970.

[8] G. Cottafava, O. D'Antona - Un programma per il disegno interattivo di grafi - C.I.O.C., Internal Rept. n° 24, Bologna 1973.

[9] H. Whitney - Congruent Graphs and the Connectivity of Graphs. - American Journal of Mathematics, Vol. 54, pp. 54, pp. 150-168, 1932.

[10] W. T. Tutte - On Hamiltonian Circuits - J. London Math. Soc., Vol. 21, pp. 98-101, 1946.

[11] W. T. Tutte - The Enumerative Theory of Planar Maps - A Survey of Combinatorial Theory, Srivastava Ed., Ch. 37, pp. 437-448, North-Holland, 1973.

[12] W. L. Engl, D. A. Mlynski - Embedding a Graph in a plane with certain constraints - IEEE Trans. on Circuit Theory, CT-17, pp. 250-252, 1970.

THE OPTIMAL CONTROL OF HAMILTONIAN SYSTEMS

EARL R. BARNES
IBM Watson Research Center
Yorktown Heights, New York 10598

ABSTRACT

We consider the optimal control of a dynamical system whose motion is described by a Hamiltonian system of differential equations and which moves so as to minimize a certain quantity such as energy or action. For such a system, there will in general be many controls satisfying Pontryagin's necessary optimality conditions. We show that the adjoint variables in these conditions can be given explicitly in terms of the state variables. This makes possible the construction of controls satisfying the necessary conditions. Our main result is a sufficiency theorem showing when such a control is optimal.

1. INTRODUCTION

Consider a control system which, for a given control u(t), moves so as to minimize a certain "action" integral. The motion will then satisfy a set of differential equations, the Euler equations for the action integral. We assume that the Euler equations are equivalent to a Hamiltonian system of the form

$$\begin{aligned} \dot{x} &= A(t)x + f(y) \\ \dot{y} &= -A^T(t)y - u(t)g(x), \qquad 0 \leq t \leq \ell , \end{aligned} \tag{1.1}$$

with boundary conditions

$$\begin{aligned} y(0) &= Mx(0) \\ y(\ell) &= -Nx(\ell). \end{aligned} \tag{1.2}$$

x(t) and y(t) are n dimensional vectors describing the state of the system at time t. u(t) is a scalar control function. It is assumed to be measurable and to satisfy constraints of the form

$$a \leq u(t) \leq b \tag{1.3}$$

where a and b are given positive constants. A(t) is an nxn continuous matrix and $A^T(t)$ is its transpose[†]. The functions f and g are vector valued with continuously differentiable components $f_1(y), \ldots, f_n(y)$ and $g_1(x), \ldots, g_n(x)$ respectively. These components are assumed to be homogeneous of degree 1 and to satisfy the conditions

$$\frac{\partial f_i(y)}{\partial y_j} = \frac{\partial f_j(y)}{\partial y_i} , \quad \frac{\partial g_i(x)}{\partial x_j} = \frac{\partial g_j(x)}{\partial x_i} , \quad i, j=1,\ldots,n , \tag{1.4}$$

for all x and y in E_n, the Euclidean space of dimension n. These conditions imply that there exist continuously differentiable functions F(y) and G(x) such that

$$\frac{\partial F(y)}{\partial y_i} = f_i(y) \text{ and } \frac{\partial G(x)}{\partial x_i} = g_i(x), \quad i=1,\ldots,n. \tag{1.5}$$

[†] The i,j-th entry in A(t) will be denoted by $a_{ij}(t)$.

Since the f_i and g_i are homogeneous of degree 1 we have, by Euler's formula,

$$f_i(y) = \sum_{j=1}^{n} y_j \frac{\partial f_i(y)}{\partial y_j} \quad \text{and} \quad g_i(x) = \sum_{j=1}^{n} x_j \frac{\partial g_i(x)}{\partial x_j} \tag{1.6}$$

for each i. Making use of the conditions (1.4) and (1.6), it follows readily that

$$F(y) = \frac{1}{2} \sum_{i=1}^{n} y_i f_i(y) \quad \text{and} \quad G(x) = \frac{1}{2} \sum_{i=1}^{n} x_i g_i(x) \tag{1.7}$$

satisfy the conditions (1.5), and when we refer to F and G below these are the functions we have in mind. Note that F and G are homogeneous of degree 2.

Finally, we assume that the function f(y) is monotone strictly increasing. That is, we assume that

$$(f(y_2) - f(y_1)) \cdot (y_2 - y_1) \geq 0$$

for all $y_1, y_2 \in E_n$, with equality if and only if $y_1 = y_2$. This condition implies that F(y) is strictly convex. The simplest examples of functions satisfying the conditions we have prescribed for f and g are

$$f(y) = Dy, \quad g(x) = Cx$$

where C and D are nxn symmetric matrices with D positive definite.

The matrices M and N appearing in the boundary conditions (1.2) are nxn and symmetric. These boundary conditions are not the only ones for which our theory applies. The condition at t = 0 could be replaced by $x(0) = 0$ and that at $t = \ell$ by $x(\ell) = 0$ without complicating the theory.

Let h(u) be a given continuous function defined on the interval [a,b]. We wish to find a control u^o which solves the problem

$$\text{minimize} \int_0^\ell h(u(t))dt \tag{1.8}$$

where the minimization is taken over all measurable functions satisfying conditions (1.1), (1.2), and (1.3). In addition to satisfying the differential equation (1.1) we require the motion of our system to minimize an action integral. We turn now to the derivation of this action integral.

Let F* denote the convex conjugate of F. F* is defined by

$$F^*(v) = \max_y \; v \cdot y - F(y) \; . \tag{1.9}$$

Since F is strictly convex, F*(v) is finite for all $v \in E_n$.

For a given $v \in E_n$ we shall denote by y(v) the value of y for which the minimum in (1.9) is assumed. It is known (cf. [1, Sections 12 and 23]) that F* is convex, continuously differentiable, and that

$$\frac{\partial F^*(v)}{\partial v} = y(v) \; . \tag{1.10}$$

We wish also to record for later use, the fact that F* is homogeneous of degree 2.

Given F* we can recover F by the operation

$$F(y) = \max_v \; y \cdot v - F^*(v) .$$

Denoting by v(y) the solution of this maximization problem we have

$$v(y) = \frac{\partial F}{\partial y}(y) = f(y). \qquad (1.11)$$

Consider now the optimal control problem[††]

$$\text{minimize } \frac{1}{2}\, x(0)\cdot Mx(0) + \frac{1}{2}\, x(\ell)\cdot Nx(\ell) + \int_0^{\ell} \{F^*(v(t)) - u(t)G(x(t))\}dt \qquad (1.12)$$

$$\text{subject to } \dot{x}(t) = A(t)\, x(t) + v(t).$$

According to Pontryagin's maximum principle, in order that $v^o(t)$ be optimal for the problem (1.12), it is necessary that there exist a function y^o satisfying

$$\begin{aligned} \dot{y}^o(t) &= -A^T(t)\, y^o(t) - u(t)\, \frac{\partial G(x^o(t))}{\partial x} \\ y^o(0) &= Mx^o(0),\ y^o(\ell) = -Nx^o(\ell) \end{aligned} \qquad (1.13)$$

such that

$$\max_v \{y^o(t)\cdot v - F^*(v)\} = y^o(t)\cdot v^o(t) - F^*(v^o(t)) \qquad (1.14)$$

for almost all $t \in [0,\ell]$. x^o is the solution of the differential equation in (1.12) corresponding to v^o.

(1.14) shows that

$$v^o(t) = v(y^o(t)) = f(y^o(t)) \quad \text{(by (1.11))}.$$

$x^o(t)$ satisfies

$$\dot{x}^o(t) = A(t)\, x^o(t) + v^o(t) = A(t)\, x^o(t) + f(y^o(t)).$$

This equation, and the equation (1.13), are the variational equations associated with the optimal control problem (1.12). Since these equations are the same as (1.1) we have shown that these latter are the variational equations for the problem (1.12) as claimed.

Pontryagin's maximum principle is only a necessary condition for optimality. Therefore, a solution of the equations (1.1), (1.2) need not correspond to a solution of the problem (1.12). In fact, the equations (1.1), (1.2) may have a solution at the same time that the problem (1.12) has no solution. A control u will be said to be admissible if in addition to (1.3) it is such that the problem (1.12) has a solution. That is, an admissible control is one for which our system has a finite response. There seems to be no mathematically simple way to improving this constraint on a control. So in our search for an optimal control we temporarily ignore this requirement. The problem of determining an optimal control is then reduced to determining proper values for n+1 variables. If these can be chosen such that the resulting control is admissible, (this happens in our example problems) then our problem is solved, for, as we shall show, Pontryagin's maximum principle, together with admissibility, is sufficient for optimality.

[††] Here u is fixed and v is the unconstrained control.

2. THE OPTIMAL CONTROL

Let u^o be the solution of the problem formulated in (1.1), (1.2), (1.3) and (1.8), and let x^o and y^o denote the corresponding solution of the boundary value problem (1.1), (1.2). Then, by Pontryagin's maximum principle, there exist adjoint variables $\psi = (\psi_1, \dots, \psi_n)$ (corresponding to x) and $\eta = (\eta_1, \dots, \eta_n)$ (corresponding to y) satisfying the conditions

$$\begin{aligned} \dot{\psi}_i &= -\sum_{j=1}^{n} a_{ji}\psi_j + u^o(t) \sum_{j=1}^{n} \frac{\partial g_j(x^o(t))}{\partial x_i}\, \eta_j \\ \dot{\eta}_i &= \sum_{j=1}^{n} a_{ij}\eta_j - \sum_{j=1}^{n} \frac{\partial f_j(y^o(t))}{\partial y_i}\, \psi_j , \quad i=1,\dots,n, \end{aligned} \qquad (2.1)$$

and a constant $\nu_o \geq 0$ such that

$$\begin{aligned} &\max_{a \leq u \leq b} \{-u\eta(t) \cdot g(x^o(t)) - \nu_o h(u)\} \\ &\quad = -u^o(t)\, \eta(t) \cdot g(x^o(t)) - \nu_o h(u^o(t)) . \end{aligned} \qquad (2.2)$$

Moreover, the vectors $(\psi(0), \eta(0))$ and $(\psi(\ell), \eta(\ell))$ are normal to the manifolds $y = Mx$ and $y = -Nx$ respectively.

Consider the choices $\psi_i^o(t) = -y_i^o(t)$, $\eta_i^o(t) = x_i^o(t)$, $i=1,\dots,n$, for the functions $\psi(t)$ and $\eta(t)$. We have, by (1.1), (1.4), and (1.6),

$$\begin{aligned} \dot{\psi}_i^o &= \sum_{j=1}^{n} a_{ji} y_j^o + u^o(t) g_i(x^o) \\ &= -\sum_{j=1}^{n} a_{ji}\psi_j^o + u^o(t) \sum_{j=1}^{n} \frac{\partial g_i}{\partial x_j}(x^o)\, x_j^o \\ &= -\sum_{j=1}^{n} a_{ji}\psi_j^o + u^o(t) \sum_{j=1}^{n} \frac{\partial g_j(x^o)}{\partial x_i}\, \eta_j^o , \end{aligned}$$

$i=1,\dots,n$. Similarly,

$$\dot{\eta}_i^o = \sum_{j=1}^{n} a_{ij}\eta_j - \sum_{j=1}^{n} \frac{\partial f_j(y^o)}{\partial y_i}\, \psi_j, \quad i=1,\dots,n.$$

Thus ψ^o and η^o satisfy the equations (2.1). Moreover, the vectors $(\psi^o(0), \eta^o(0))$, $(\psi^o(\ell), \eta^o(\ell))$ are normal to the manifolds $y = Mx$ and $y = -Nx$ respectively. To see this note that the vector (x,y) is in the manifold $y = Mx$ if and only if the vector $(x,y)^T$ is in the null space of the matrix operator $[-M,I]$. I is the nxn identity matrix. Moreover, the vector $(\psi^o(0), \eta^o(0))^T$ is normal to the null space of $[-M,I]$ if and only if it belongs to the range of the transposed matrix

$$\begin{bmatrix} -M \\ I \end{bmatrix} . \qquad (2.3)$$

Here we use the symmetry of M. But $(\psi^o(0), \eta^o(0))^T$ is certainly in the range of (2.3) since the equation

$$\begin{bmatrix} -M \\ I \end{bmatrix} z = \begin{pmatrix} \psi^o(0) \\ \eta^o(0) \end{pmatrix}$$

has the solution $z = x^o(0)$. Similarly we can show that $(\psi^o(\ell), \eta^o(\ell))^T$ is normal to the manifold $y = -Nx$.

These normality conditions are trivial to check when the boundary conditions on our system are of the form $x(0) = 0$ or $x(\ell) = 0$.

Let us now assign trial values to the variables ν_o and $x^o(0)$. Using the value of η^o just determined, we can then compute trial values of $u^o(t)$ from (2.2) for each value of t for which $x^o(t)$ is known. This allows us to solve the system (1.1) forward on $[0,\ell]$ using the initial value $y^o(0) = Mx^o(0)$. Having done this we have produced a trial control $u^o(t)$. If it happens that we have chosen ν_o and $x^o(0)$ in such a way that the condition $y^o(\ell) = -Nx^o(\ell)$ is satisfied, $u^o(t)$ satisfies all the conditions of Pontryagin's maximum principle. If in addition ν_o and $x^o(0)$ have been chosen such that $\nu_o > 0$ and u^o is admissible, then, as the following theorem shows, u^o is optimal.

Theorem. Let an admissible control u^o satisfying the conditions of Pontryagin's maximum principle be constructed as above and assume that $\nu_o > 0$. Let $x^o(t)$, $y^o(t)$ denote the corresponding solution of (1.1), (1.2). Then the pair $x^o(t)$, $v^o(t) = v(y^o(t))$ is a solution of the problem (1.12) with $u = u^o$, and

$$\int_0^\ell h(u^o(t))dt \leq \int_0^\ell h(u(t))dt \tag{2.4}$$

for any admissible control u.

Proof. In order to prove that the pair $x^o(t)$, $v^o(t) = v(y^o(t))$ is a solution of the problem (1.12) we shall prove that for any triple $u(t)$, $x(t)$, $y(t)$ satisfying (1.1), (1.2), the pair $x(t)$, $v(t) = v(y(t))$ reduces the expression in (1.12) to zero. The conclusion will then follow, because in particular, the solution of (1.12) corresponding to $u = u^o$ satisfies (1.1) and (1.2).

Let $u(t)$, $x(t)$, $y(t)$ satisfy (1.1) and (1.2). We then have

$$\begin{aligned} -x(0) \cdot Mx(0) - x(\ell) \cdot Nx(\ell) &= \int_0^\ell \frac{d}{dt} (x \cdot y)\, dt \\ &= \int_0^\ell \{(Ax + f(y)) \cdot y + (-A^T y - ug(x)) \cdot x\}\, dt \\ &= \int_0^\ell \{y \cdot f(y) - ux \cdot g(x)\}\, dt \\ &= \int_0^\ell \{2F^*(v) - 2u \cdot G(x)\}\, dt \end{aligned}$$

where $v(t) = v(y(t)) = f(y(t))$ and

$$y(t) = \frac{\partial F^*}{\partial v}(v(t))$$

so that

$$\int_0^\ell y \cdot f(y)dt = \int_0^\ell \frac{\partial F^*}{\partial v}(v(t)) \cdot v(t)dt$$

$$= \int_0^\ell 2F^*(v(t))dt$$

by the degree 2 homogeneity of F*. This proves the first conclusion in the theorem. To prove (2.4) note that for any triple u(t), x(t), y(t) satisfying (1.1), (1.2), with u(t) admissible, we have, by what was just shown,

$$0 = \frac{1}{2} x(0) \cdot Mx(0) + \frac{1}{2} x(\ell) \cdot Nx(\ell + \int_0^\ell \{F^*(v(t)) - u(t)\ G(x(t))\}dt$$

$$- \frac{1}{2} x^o(0) \cdot Mx^o(0) - \frac{1}{2} x^o(\ell) \cdot Nx^o(\ell) - \int_0^\ell \{F^*(v^o(t)) - u^o(t) G(x^o(t))\}dt$$

$$= - \int_0^\ell (u(t) - u^o(t))\ G(x^o(t))\ dt$$

$$+ \frac{1}{2} x(0) \cdot Mx(0) + \frac{1}{2} x(\ell) \cdot Nx(\ell) + \int_0^\ell \{F^*(v(t)) - u(t)\ G(x(t))\}dt$$

$$- \frac{1}{2} x^o(0) \cdot Mx^o(0) - \frac{1}{2} x^o(\ell) \cdot Nx^o(\ell) - \int_0^\ell \{F^*(v^o(t)) - u(t) G(x^o(t))\}dt$$

$$\leq - \int_0^\ell (u(t) - u^o(t))\ G(x^o(t))\ dt\ .$$

This inequality follows from the minimizing property of the pair x(t), v(t) corresponding to u(t). It shows that

$$\nu_o \int_0^\ell \{h(u(t)) - h(u^o(t))\}dt$$

$$\geq \int_0^\ell \{\nu_o h(u(t)) - \nu_o h(u^o(t)) + 2(u(t) - u^o(t))G(x^o(t))\}dt$$

$$= \int_0^\ell \{-2u^o(t)\ G(x^o(t)) - \nu_o h(u^o(t))\}dt$$

$$- \int_0^\ell \{-2u(t)\ G(x^o(t)) - \nu_o h(u(t))\}dt$$

$$= \int_0^\ell \{-2u^o(t)\ \eta^o(t) \cdot g(x^o(t)) - \eta_o h(u^o(t))\}dt$$

$$- \int_0^\ell \{-2u(t)\ \eta^o(t) \cdot g(x^o(t)) - \nu_o h(u(t))\}dt$$

$$\geq 0$$

by (2.2). This completes the proof of the theorem since $\nu_o > 0$.

3. AN EXAMPLE

Consider the problem of determining the shape of the column having minimum weight, fixed length ℓ, and similar cross sections, and supporting a load P in an unbuckled state. We also require that the area of each cross section be $\leq$ H and $\geq$ h for some given positive constants h and H. A similar problem is solved in [2] without these constraints on the area of cross sections.

Let distance along the column be measured by ξ. Assume that the column is not twisted and let $w(\xi)$ denote the deflection at ξ. Then, according to [2], w satisfies

$$(E\,I(\xi)w_{\xi\xi})_{\xi\xi} + Pw_{\xi\xi} = 0, \quad 0 \leq \xi \leq \ell, \tag{3.1}$$

with boundary conditions

$$\begin{aligned} &w(0) = w_{\xi}(0) = 0 \\ &(w_{\xi}(\ell) = Pw_{\xi}(\ell) + (E\,I(\ell)w_{\xi\xi}(\ell))_{\xi} = 0 \end{aligned} \tag{3.2}$$

for the case where the column is clamped at $\xi = 0$ and hinged at $\xi = \ell$, or

$$\begin{aligned} &w(0) = w_{\xi}(0) = 0 \\ &w(\ell) = w_{\xi\xi}(\ell) = 0 \end{aligned} \tag{3.3}$$

for the case of a column clamped at $\xi = 0$ and free at $\xi = \ell$.

E is Young's modulus of the column material and $I(\xi)$ is the moment of inertia of a cross section about a line through its centroid normal to the plane of the deflected column. $I(\xi)$ is related to the area $A(\xi)$ of the cross section at ξ by

$$I(\xi) = \alpha A^{2}(\xi)$$

where α is a proportionality factor determined by the shape of the cross sections.

If we make the change of variables $t = \xi$, $z(t) = A^{2}(\xi)w_{\xi\xi}(\xi)$, $\lambda = \frac{P}{E\alpha}$, the differential equation (3.1) becomes

$$\ddot{z} + \lambda A^{-2}(t)z = 0 \tag{3.4}$$

with boundary conditions

$$\begin{aligned} &\dot{z}(0) = 0 \\ &z(\ell) = 0 \end{aligned} \qquad \text{(clamped-hinged)} \tag{3.5}$$

or

$$\begin{aligned} &\dot{z}(0) + \ell^{-1}z(0) = 0 \\ &z(\ell) = 0 \end{aligned} \qquad \text{(clamped-free)} \tag{3.6}$$

corresponding to (3.1) and (3.2) respectively.

For simplicity we single out the clamped-free case for solution. Since our column is to have minimum weight we may assume that it will support no load larger than P. P is then the smallest eigenvalue of the problem (3.1), (3.2) and λ is the smallest eigenvalue of the problem (3.4), (3.5). We therefore have

$$\lambda = \min_{z(\ell)=0} \frac{\int_0^{\ell} \dot{z}^{2}(t)\,dt}{\int_0^{\ell} A^{-2}(t)z^{2}(t)\,dt} ,$$

and the solution of this minimization problem is also a solution of the problem

$$\underset{z(\ell)=0}{\text{minimize}} \int_0^{\ell} \{\dot{z}^{2}(t) - \lambda A^{-2}(t)z^{2}(t)\}dt. \tag{3.7}$$

The weight of our column is proportional to

$$\int_0^{\ell} A(t)dt \tag{3.8}$$

and our problem is therefore to minimize (3.8) subject to (3.4) and the constraint

$$h \leq A(t) \leq H \tag{3.9}$$

on A. An admissible A must also be such that the problem (3.7) has a solution.

If we let $x(t) = z(t)$, $y(t) = \dot{z}(t)$, $u(t) = A^{-2}(t)$, then this minimization problem can be formulated as

$$\text{minimize} \int_0^{\ell} u^{-1/2}(t)dt$$

subject to the conditions

$$\text{(i)} \quad \begin{aligned} \dot{x} &= y &&, \ x(\ell) = 0 \\ \dot{y} &= -u(t)\lambda x &&, \ y(0) = 0 , \end{aligned} \tag{3.10}$$

(ii) $a \leq u(t) \leq b$,

(iii) The problem

$$\text{minimize} \int_0^{\ell} \{\tfrac{1}{2} v^2 - u(t)\lambda x^2\}dt$$

$$\dot{x} = v, \ x(\ell) = 0,$$

has a solution.

To ensure that the class of admissible controls is not empty, and that our problem is nontrivial, we assume that the control $u(t) \equiv a$ is admissible and that the control $u(t) \equiv b$ is not admissible. These conditions imply that the column corresponding to $A(\xi) \equiv H$ will support our load while the one corresponding to $A(\xi) \equiv h$ will not. If the column $A(\xi) \equiv h$ could support our load, it would of course be optimal and our problem would be trivial. A necessary and sufficient condition that a control u(t) be admissible is that the boundary value problem

$$\ddot{x} + u(t)\lambda x = 0$$

$$\dot{x}(0) = x(\ell) = 0$$

have no nonzero solution which vanishes at a point in $[0,\ell)$. This says that a control u(t) is admissible if and only if λ is less than or equal to the first eigenvalue of this boundary value problem. Thus the class of admissible controls are precisely the ones which will support any load < λ in an unbuckled state. The control $u(t) \equiv a$ is therefore admissible if and only if $\cos\sqrt{a\lambda}\, t > 0$ for $0 \leq t < \ell$. We shall assume that

$$\sqrt{a\lambda}\, \ell < \frac{\pi}{2} \tag{3.11}$$

and that

$$\pi > \sqrt{b\lambda}\, \ell > \frac{\pi}{2} \quad . \tag{3.12}$$

(3.12) ensures that the control $u(t) \equiv b$ is not admissible.

The differential equation (3.10i) is homogeneous and therefore a solutions is unique only up to a scalar factor. To pin down a unique solution we shall impose the condition $x(0) = 1$ on solutions.

We turn now to the construction of an optimal control u^{o}. According to (2.2) (with $\eta = x^{o}$) u^{o} must satisfy the condition

$$\max_{a\le u\le b}\ \{-u\cdot\tfrac{1}{2}\,(x^o(t))^2 - \nu_o u^{-1/2}\}$$

$$= -u^o(t)\cdot\tfrac{1}{2}(x^o(t))^2 - \nu_o(u^o(t))^{-1/2}$$

for each $t \in [0,\ell]$. It follows that

$$u^o(t) = \begin{cases} a \text{ if } \nu_o^2(x^o(t))^{-4/3} \le a \\ \nu_o^2(x^o(t))^{-4/3} \text{ if } a \le \nu_o^2(x^o(t))^{-4/3} \le b \\ b \text{ if } \nu_o^2(x^o(t))^{-4/3} \ge b \end{cases} \tag{3.13}$$

for some constant $\nu_o \ge 0$, provided an optimal control u^o exists. But the existence of an optimal control follows immediately from our assumptions (3.11) and (3.12). To see this, consider the solution of (3.10i) with $x(0) = 1$ and $u = u^o$ given by (3.13). If $\nu_o = 0$ we have $u^o(t) \equiv a$ and $x(t) = \cos\sqrt{a\lambda}\ t$. (3.11) shows that $x(\ell) > 0$. On the other hand, if we take $\nu_o = \sqrt{b}$, then, as the reader can easily verify, $u^o(t) \equiv b$ so that $x(t) = \cos\sqrt{b\lambda}\ t$. Condition (3.12) therefore implies that $x(\ell) < 0$. Now it is clear that solutions of (3.10i) depend continuously on the parameter ν_o. It therefore follows that there is a number ν_o, $0 < \nu_o < \sqrt{b}$ such that $x(\ell) = 0$. The u^o corresponding to this ν_o satisfies all the conditions of Pontryagin's maximum principle and is admissible since the corresponding x is positive on $[0,\ell)$. It follows from the theorem of Section 2 that this control is optimal.

We shall now derive an equation for ν_o. There are two cases to consider. The first is where $a \le \nu_o^2 < b$. In this case let t_1 be the smallest value of t for which $\nu_o^2(x^o(t))^{-4/3} = b$. Then for $0 \le t \le t_1$ we have $u^o(t) = \nu_o^2(x^o(t))^{-4/3}$. Substituting this into (3.10i) we find that, for $x = x^o$ and $y = y^o$,

$$\frac{d}{dt}\ \{3\lambda\nu_o^2\ x^{2/3} + y^2\ \} = 0$$

which implies that

$$3\lambda\ \nu_o^2\ x^{2/3}(t) + \dot{x}^2(t) = 3\lambda\ \nu_o^2 \tag{3.14}$$

for $0 \le t \le t_1$. Let $\theta(t)$ be a function defined by the requirement that $(\sin\theta(t))^3 = x(t)$. Substituting this into (3.14) we obtain the differential equation

$$\sin^2\theta\ \frac{d\theta}{dt} = -\sqrt{3\lambda}\ \ \nu_o$$

for θ. We may assume that $\theta(0) = \frac{\pi}{2}$.

We then have

$$\int_{\frac{\pi}{2}}^{\theta(t_1)} \sin^2\theta\ d\theta = -\sqrt{\frac{\lambda}{3}}\ \nu_o t_1$$

or

$$\frac{\theta(t_1)}{2} - \frac{\sin 2\theta(t_1)}{4} - \frac{\pi}{2} = -\sqrt{\frac{\lambda}{3}}\ \nu_o t_1\ . \tag{3.15}$$

Now since $\nu_o^2(x^o(t_1))^{-4/3} = b$, and $x(t_1) = \sin\theta(t_1))^3$, we have

$$\theta(t_1) = \sin^{-1}\left(\frac{\nu_o^2}{b}\right)^{1/4}\ .$$

Substituting this into (3.15) gives

$$t_1 = \sqrt{\frac{3}{\lambda}}\,(2\nu_o)^{-1}\left\{\pi + \left(\frac{\nu_o^2}{b}\right)^{1/4}\sqrt{1 - \left(\frac{\nu_o^2}{b}\right)^{1/2}} - \sin^{-1}\left(\frac{\nu_o^2}{b}\right)^{1/4}\right. \tag{3.16}$$

On the interval $[t_1, \ell]$ we have $u^o(t) = b$ and, by (3.10i), the motion $x = x^o$, $y = y^o$ satisfies

$$\frac{d}{dt}\,\{b\lambda x^2 + y^2\} = 0$$

which implies that

$$b\lambda x^2(t) + \dot{x}^2(t) = b\lambda k^2$$

for $t_1 \leq t \leq \ell$, where

$$k = \left\{\left(\frac{\nu_o^2}{b}\right)^{3/2} + \frac{3}{b}\,\nu_o^2\left(1 - \left(\frac{\nu_o^2}{b}\right)^{1/2}\right)\right\}^{1/2} .$$

Since $x^o(\ell) = 0$, this equation implies that

$$\frac{1}{\sqrt{b\lambda}} \int_{\left(\frac{\nu_o^2}{b}\right)^{3/2}}^{0} \frac{dx}{\sqrt{k^2 - x^2}} = -\int_{t_1}^{\ell} dt .$$

Solving this equation for t_1 we find that

$$t_1 = \ell - \frac{1}{\sqrt{b\lambda}}\,\sin^{-1}\left(\frac{\nu_o^2}{bk^2}\right)^{3/2}$$

Equating this to the value to t_1 given by (3.16) we obtain an equation for ν_o. In this equation $\sin^{-1}$ is to be given its smallest positive value. Then if there is no solution in the interval $[\sqrt{a}, \sqrt{b})$ we must have $0 < \nu_o < \sqrt{a}$. This is the second case to consider. We leave the problem of deriving an equation for ν_o in this case to the reader.

Remark. In connection with this work we wish to mention that of Athans, Falb, and Lacoss [3], [4]. They were among the first to identify classes of problems where the adjoint variables are explicit functions of the state variables. In their problems the controls appeared linearly.

REFERENCES

[1] R. T, Rockafellar, Convex Analysis, Princeton University Press, Princeton, N. J., 1969.

[2] I. Tadjbakhsh and J. B. Keller, Strongest Columns and Isoperimetric Inequalities for Eigenvalues", ASME Journal of Applied Mechanics, Vol. 29, Series E, 1962, pp. 159-164.

[3] M. Athans, P. L. Falb, R. T. Lacoss, "On Optimal Control of Self-Adjoint Systems", Proceedings of the 1963 Joint Automatic Control Conference, Minneapolis, Minn., June 21-23, 1963, pp. 113-120.

[4] ___________, "Time-, Fuel-, and Energy-Optimal Control of Nonlinear Norm-Invariant Systems", IEEE Transactions on Automatic Control, Vol. AC-8, July, 1963, pp. 196-202.

UNCONSTRAINED METHODS IN OPTIMIZATION1)

O. L. MANGASARIAN2)
Computer Sciences Department, University of Wisconsin, Madison,
Wisconsin 53706

ABSTRACT

Some simple ideas are presented which make stationary points of inequality and equality constrained optimization problems equivalent to unconstrained solutions of a system of nonlinear equations. These ideas generalize the methods of unconstrained or augmented Lagrangians. A parametrically superlinearly convergent class of algorithms that includes that of Powell and Hestenes is proposed.

We shall consider in this paper the problem

$$\text{minimize } f(x), \text{ subject to } g(x) \leqq 0 \tag{1}$$

where $f:R^n \to R$ and $g:R^n \to R^m$. For the sake of simplicity we have excluded equality constraints which can be handled by minor modifications of the methods described here. We shall be concerned with stationary points of (1), that is points x in R^n which, for some $u \in R^m$, satisfy the Kuhn-Tucker conditions

$$\nabla f(x) + \nabla g(x)u = 0 \tag{2a}$$

$$ug(x) = 0,\ g(x) \leqq 0,\ u \geqq 0 \tag{2b}$$

where $\nabla f(x)$ is the gradient of f at x and $\nabla g(x)$ is the $n \times m$ Jacobian of g at x. We observe immediately that if the conditions (2b) can be replaced by an equivalent system of m equations without an increase in the number of variables then the Kuhn-Tucker conditions become equivalent to a system of $n + m$ equations in $n + m$ unknowns. That this can be done follows from the following simple but key lemma which was first proved in [9, lemma 2.7]

1. Lemma Let $y \in R$, $g \in R$ and let θ be an injective function from R into R (that is $\theta(a) = \theta(b)$ implies $a = b$) with $\theta(0) = 0$. Then

$$yg = 0,\ g \leqq 0,\ y \geqq 0 \Leftrightarrow \theta(g+y)_+ - \theta(y) = 0$$

where
$$\theta(z)_+ = \begin{cases} \theta(z) & \text{if } z \geqq 0 \\ 0 & \text{if } z < 0 \end{cases}.$$

Proof
$$\theta(g+y)_+ - \theta(y) = 0 \Leftrightarrow \left\langle \begin{matrix} g+y \geqq 0,\ g+y = y \\ \text{or} \\ g+y < 0,\ y = 0 \end{matrix} \right\rangle \Leftrightarrow \left\langle \begin{matrix} g=0,\ y\geqq 0 \\ \text{or} \\ g<0,\ y=0 \end{matrix} \right\rangle$$
$$\Leftrightarrow yg = 0,\ g \leqq 0,\ y \geqq 0 . \qquad \text{QED}$$

The following theorem, a generalization of [9, theorem 2.5], establishes the equivalence of the Kuhn-Tucker conditions to a system of $n + m$ equations in $n + m$ unknowns.

2. Theorem: Let θ be an injective function from R into R with $\theta(0) = 0$, and let ϕ be a strictly increasing surjective function from

---

1) Research supported by NSF Grant GJ35292

2) Computer Sciences Department, University of Wisconsin, 1210 West Dayton Street, Madison, Wisconsin 53706

R into R (that is ϕ maps R onto R) with $\phi(0) = 0$. Then for any $\alpha > 0$

$$\left\langle \begin{array}{l} \nabla f(x) + \nabla g(x)u = 0 \\ \\ ug(x)=0, g(x) \leqq 0, u \geqq 0 \end{array} \right\rangle \begin{array}{c} u_j=\phi(y_j) \\ \Longleftrightarrow \\ j=1,\ldots,m \end{array} \left\langle \begin{array}{l} \nabla f(x)+\sum_{j=1}^{m} \phi(\alpha g_j(x)+y_j)_+ \nabla g_j(x)=0 \\ \\ \theta(\alpha g_j(x)+y_j)_+ - \theta(y_j)=0 \\ \qquad j=1,\ldots,m \end{array} \right\rangle \quad (3)$$

Proof: By lemma 1 and $\phi(\alpha g_j(x)+y_j)_+ = \phi(y_j)_+ = \phi(y_j) = u_j$ for $g_j(x) = 0$ and $\phi(\alpha g_j(x)+y_j)_+ = \phi(\alpha g_j(x))_+ = 0 = \phi(y_j) = u_j$ for $g_j(x) < 0$. QED

If we let $\alpha\theta(z) = \phi(z) = z$ then equations (3) become the gradient of Rockafellar's augmented Lagrangian [12] which has been intensively investigated recently [1,2,6,9]. Unfortunately equations (3) are not differentiable globally if $\theta'(0) \neq 0$ as is the case for Rockafellar's Lagrangian, although they are differentiable near a solution $(\bar{x},\bar{y})$ if we assume that strict complementarity holds, that is $\bar{y}_j+|g_j(\bar{x})|>0, j=1,\ldots,m$. In [9] by using $\alpha\theta(z)=\phi(z)=|z|^t z$, $t > 0$, equations (3) become differentiable globally. However the Jacobian of (3) becomes singular at the solution if there are inactive constrants. This difficulty was avoided in [9] by further augmenting the Lagrangian. We propose another way here to make the equations (3) differentiable globally while maintaining nonsingularity of the Jacobian by symmetrizing the last m equations of (3) with respect to $g_j(x)$ and y_j . This will have the advantage over [9] in that the ingenious method of Powell [11] and Hestenes [8] can be extended to inequalities while still maintaining global differentiability and a rate of convergence that is faster than any linear rate. We first symmetrize lemma 1 by interchanging the roles of g and y and exploiting the antisymmetry of θ .

3 Lemma Let $y \in R$, $g \in R$ and let θ be a strictly increasing odd function from R into R with $\theta(0) = 0$. Then

$$yg = 0,\ g \leqq 0,\ y \geqq 0 \Leftrightarrow \theta(|g+y|) + \theta(g) - \theta(y) = 0 .$$

Proof: If we let $h = -g$, then it is sufficient to establish that

$$yh = 0,\ h \geqq 0\ \ y \geqq 0 \ \Leftrightarrow\ \theta(|h-y|) - \theta(h) - \theta(y) = 0$$

(⇒) Either $y = 0$ or $h = 0$. If $y = 0$, then $\theta(|h-y|) - \theta(h) - \theta(y) = \theta(|h|) - \theta(h) = 0$. If $h = 0$ a similar argument goes through with h and y interchanged.

(⇐) (a) To show that $h \geqq 0$, assume the contrary that $h < 0$. Then $0 \leqq \theta(|y-h|) = \theta(h) + \theta(y) < \theta(y)$, from which it follows that $y > 0$ and $y > |y-h| = y - h$. The last inequality contradicts $h < 0$.

(b) To show that $y \geqq 0$, interchange the roles of y and h in (a).

(c) To show that $yh = 0$, assume the contrary that $y > 0$ and $h > 0$. Without loss of generality assume that $h \geqq y$, then $\theta(|h-y|) = \theta(h-y) < \theta(h) < \theta(h) + \theta(y)$. This contradicts $\theta(|h-y|) - \theta(h) - \theta(y) = 0$. QED.

The following equivalence theorem follows from lemma 3 in the same way as theorem 2 followed from lemma 1.

4 Theorem Let θ be a strictly increasing odd function from R into R with $\theta(0) = 0$ and let ϕ be a strictly increasing surjective function from R into R with $\phi(0) = 0$. Then for any $\alpha > 0$

$$\left.\begin{array}{l}\nabla f(x)+\nabla g(x)u=0\\ \\ ug(x)=0,g(x)\leq 0,u\geq 0\end{array}\right\rangle \Leftrightarrow \left\langle\begin{array}{l} L_1(x,y,\alpha)=\nabla f(x)+\sum\limits_{j=1}^{m} \phi(\alpha g_j(x)+y_j)_+\nabla g_j(x)=0\\ \\ L_2(x,y,\alpha)=\theta(|\alpha g_j(x)+y_j|)+\theta(\alpha g_j(x))-\theta(y_j)=0\\ \qquad\qquad j=1,\ldots,m\end{array}\right\rangle \tag{4}$$

and $u_j = \phi(y_j), j=1,\ldots,m$.

Again note that if we set $2\alpha\theta(z) = \phi(z) = z$ then equations (4) become the gradient of Rockafellar's Lagrangian. However we shall be concerned here mainly with functions θ and ϕ such that $\theta'(0) = \phi'(0)= 0$ in order that equations (4) be globally differentiable. The key to computational algorithms for solving (4) is the nonsingularity and the structure of its Jacobian at the solution. Let $L(x,y,\alpha) = \begin{bmatrix} L_1(x,y,\alpha) \\ L_2(x,y,\alpha)\end{bmatrix} = 0$ denote the $m + n$ equations (4) and let $(\bar{x},\bar{y})$ be a solution of (4) at which strict complementarity holds, $\bar{y}_j+|g_j(\bar{x})|>0, j=1,\ldots,m$. If ϕ and θ are differentiable on R with derivatives that vanish at the origin only, and if f and g are differentiable at $\bar{x}$, then

$$\nabla L(\bar{x},\bar{y},\alpha) = \begin{bmatrix} \begin{array}{l}[\nabla_{11}L^0(\bar{x},\bar{u})+\sum\limits_{i\in I}\alpha\phi'(\bar{y}_i)\cdot \\ \qquad \cdot\nabla g_i(\bar{x})\nabla g_i(\bar{x})^T]\end{array} & \underset{(i\in I)}{\nabla g_i(\bar{x})\phi'(\bar{y}_i)} & 0 \\ \\ \underset{(i\in I)}{\alpha(\theta'(\bar{y}_i)+\theta'(0))\nabla g_i(\bar{x})^T} & 0 & 0 \\ \\ 0 & 0 & \underset{(i\in J)}{-\theta'(-\alpha g_i(\bar{x}))-\theta'(0)} \end{bmatrix} \tag{5}$$

where $\bar{u}_j = \phi(\bar{y}_j), j=1,\ldots,m, L^0(x,u)=f(x)+\sum\limits_{j=1}^{m} u_j g_j(x)$, the classical Lagrangian, $I = \{j|g_j(\bar{x})=0\}$ and $J = \{j|g_j(\bar{x})<0\}$. It follows that by Debreu's theorem [4] (which states that $N + \alpha M^T M$ is positive definite for $\alpha \geq \bar{\alpha}$ for some $\bar{\alpha} > 0$ if $xNx > 0$ for $Mx = 0$ and $x \neq 0$), by strict complementarity, and by second order sufficiency at $(\bar{x},\bar{u})$ [5] that the upper left $n \times n$ submatrix of (5) which we denote by $\nabla_1 L_1(\bar{x},\bar{y},\alpha)$ is positive definite for $\alpha \geq \bar{\alpha}$ for some $\bar{\alpha} > 0$. This together with the linear independence of $\nabla g_i(\bar{x}), i \in I$ and $-\theta'(-\alpha g_i(\bar{x})) - \theta'(0) < 0$, $i \in J$, insure the nonsingularity of $\nabla L(\bar{x},\bar{y},\alpha)$. Henceforth we shall assume in the paper that strict complementarity, second order sufficiency and linear independence of the gradients of the active constraints hold at $(\bar{x},\bar{u})$.

An immediate algorithm for solving (4) would be a Newton or quasi-Newton algorithm [10,3] which would be locally superlinearly convergent if $\alpha \geqq \bar{\alpha}$ for some $\bar{\alpha} > 0$. A concrete realization of (4) to be solved by such methods can be obtained by setting

$$\theta(z) = \phi(z) = \frac{1}{2} z|z|, \quad \theta'(z) = \phi'(z) = |z| \tag{6}$$

Note that $\theta'(0) = \phi'(0) = 0$ and hence equations (4) are globally differentiable. Probably a more interesting and effective method would be the following one which is an asymptotic Newton method in the y-space and which for a specific $\theta(z)$ and $\phi(z)$ becomes the extension to inequalities of Powell's method [11].

5. Algorithm Choose $\alpha > 0$, $y^0 \in R^m$ and $x^0 \in R^n$ such that $L_1(x^0,y^0,\alpha) = 0$. Let (x^i,y^i) determine (x^{i+1},y^{i+1}) as follows:

(a) Determine x^{i+1} such that $L_1(x^{i+1},y^i) = 0$. If x^{i+1} is not unique, take a closest x^{i+1}, in some norm, to x^i.

(b) $y^{i+1} = y^i + \alpha L_2(x^{i+1},y^i,\alpha)$.

Note that this algorithm becomes Powell's algorithm extended to inequalities if we take $2\alpha\theta(z) = \phi(z) = z$. However we shall propose another choice for $\theta(z)$ and $\phi(z)$ to use in algorithm 5 which will make $L(x,y,\alpha)$ differentiable globally. Before doing that however, we consider the local convergence of algorithm 5 and its rate of convergence under the sim-simplifying assumption that x^{i+1} is unique in step (a) of the algorithm. (The non-unique case can be handled in a manner similar to that in [9, theorem 4.10].) Let ϕ and θ be continuously differentiable on R and let f and g be twice continuously differentiable in a neighborhood of $\bar{x}$. Then by the implicit function theorem, for y^i sufficiently close to $\bar{y}$ and $\alpha \geqq \bar{\alpha} > 0$, algorithm 5 is equivalent to

$$y^{i+1} = y^i + \alpha L_2(e(y^i),y^i,\alpha) \tag{7}$$

where e is a differentiable function from R^m into R^n satisfying $L_1(e(y),y,\alpha) = 0$ for all y in a neighborhood of $\bar{y}$. Consider now the mapping $G(y) = y + \alpha L_2(e(y),y,\alpha)$ underlying the iteration (7). By using $\nabla_1 L_1(\bar{x},\bar{y},\alpha)\nabla e(\bar{y}) + \nabla_2 L_1(\bar{x},\bar{y},\alpha) = 0$ and (5) we obtain that

$$\begin{aligned}
\nabla G(\bar{y}) &= I + \alpha\nabla_1 L_2(\bar{x},\bar{y},\alpha)\nabla e(\bar{y}) + \alpha\nabla_2 L_2(\bar{x},\bar{y},\alpha) \\
&= I - \alpha(\nabla_1 L_2(\bar{x},\bar{y},\alpha)\nabla_1 L_1(\bar{x},\bar{y},\alpha)^{-1}\nabla_2 L_1(\bar{x},\bar{y},\alpha) - \nabla_2 L_2(\bar{x},\bar{y},\alpha)) \\
&= I - \alpha \begin{bmatrix} \alpha(\theta'(\bar{y}_I)+\theta'(0))\nabla g_I(\bar{x})^T[\nabla_{11}L^0(\bar{x},\bar{u}) + \nabla g_I(\bar{x})(\alpha\phi'(\bar{y}_I))\nabla g_I(\bar{x})^T]^{-1}\nabla g_I(\bar{x})\phi'(\bar{y}_I) & 0 \\ 0 & \theta'(-\alpha g_J(\bar{x}))+\theta'(0) \end{bmatrix}
\end{aligned}$$

where $\theta'(\bar{y}_I) + \theta'(0)$ denotes a diagonal matrix with diagonal elements $\theta'(y_i) + \theta'(0)$, $i \in I$. Similarly $\phi'(\bar{y}_I)$ and $\theta'(-\alpha g_J(\bar{x})) + \theta'(0)$ are also diagonal matrices. The matrix $\nabla g_I(\bar{x})$ has as its columns $\nabla g_i(\bar{x})$, $i \in I$. We now make use of the following extremely useful lemma which can be considered a key lemma for obtaining convergence rate results for unconstrained Lagrangians. In a lemma in [11] Powell proved a related result by using determinants.

6 Lemma Let $C(\alpha) = B^T(A+BQ(\alpha)B^T)^{-1}B$ where B is a given $n \times m$ matrix of rank m, A is an $n \times n$ matrix, $Q(\alpha)$ is a differentiable $m \times m$ matrix function on R and $A + BQ(\alpha)B^T$ is positive definite for $\alpha \geqq \bar{\alpha}$ for some $\bar{\alpha}$. Then $C(\alpha) = (Q(\alpha)+K)^{-1}$ for some constant $m \times m$ matrix K and all $\alpha \geqq \bar{\alpha}$.

Proof. Recall that the formula for differentiating the inverse of a matrix is given by $\frac{dC(\alpha)^{-1}}{d\alpha} = -C(\alpha)^{-1}\frac{dC(\alpha)}{d\alpha}C(\alpha)^{-1}$. Hence from the definition of $C(\alpha)$ we have that

$$\frac{dC(\alpha)}{d\alpha} = -B^T(A+BQ(\alpha)B^T)^{-1}B\frac{dQ(\alpha)}{d\alpha}B^T(A+BQ(\alpha)B^T)^{-1}B = -C(\alpha)\frac{dQ(\alpha)}{d\alpha}C(\alpha)$$

Hence $\frac{dC(\alpha)^{-1}}{d\alpha} = \frac{dQ(\alpha)}{d\alpha}$ and $C(\alpha)^{-1} = Q(\alpha) + K$. QED

By using this lemma in the last expression for $\nabla G(\bar{y})$ we obtain

$$\nabla G(\bar{y}) = I - \alpha \begin{bmatrix} \alpha(\theta'(\bar{y}_I)+\theta'(0))(\alpha\phi'(\bar{y}_I)+K)^{-1}\phi'(\bar{y}_I) & 0 \\ 0 & \theta'(-\alpha g_J(\bar{x}))+\theta'(0) \end{bmatrix} \tag{7}$$

If $2\alpha\theta(z) = \phi(z) = z$, as in Powell's method, then

$$\nabla G(\bar{y}) = I - \alpha \begin{bmatrix} (\alpha I+K)^{-1} & 0 \\ 0 & \frac{I}{\alpha} \end{bmatrix} = \begin{bmatrix} I - (I+\frac{K}{\alpha})^{-1} & 0 \\ 0 & 0 \end{bmatrix} \tag{7a}$$

Hence for sufficiently large α, $\rho(\nabla G(\bar{y})) \leqq \frac{2\,\|K\|}{\alpha}$

where $\rho(\nabla G(\bar{y}))$ is the spectral radius of $\nabla G(\bar{y})$. Hence by Ostrowski's point of attraction theorem [10, theorem 10.1.3] and [10,10.1.4] the sequence $\{y^i\}$ is locally convergent to $\bar{y}$ and the root convergence factor

$$R_1\{y^i\} = \limsup_{i\to\infty} \|y^i - \bar{y}\|^{\frac{1}{i}} \leqq 2\,\frac{\|K\|}{\alpha}$$

As $\alpha \to \infty$ this factor approaches zero and hence we have parametric root-superlinear convergence, the parameter being α of course.

We propose now another realization of algorithm 5 based on the following following functions

$$\phi(z) = \frac{1}{2} z|z| \qquad \phi'(z) = |z| \tag{8}$$

$$\theta(z) = \begin{cases} \dfrac{z}{\alpha} + \dfrac{1}{2\alpha^2} & \text{if } z \leqq -\dfrac{1}{\alpha} \\ \dfrac{1}{2} z|z| & \text{if } |z| < \dfrac{1}{\alpha} \\ \dfrac{z}{\alpha} - \dfrac{1}{2\alpha^2} & \text{if } z \geqq \dfrac{1}{\alpha} \end{cases}, \quad \theta'(z) = \begin{cases} \dfrac{1}{\alpha} & \text{if } z \leqq -\dfrac{1}{\alpha} \\ |z| & \text{if } |z| < \dfrac{1}{\alpha} \\ \dfrac{1}{\alpha} & \text{if } z \geqq \dfrac{1}{\alpha} \end{cases} \tag{9}$$

Note that $\theta'(0) = 0$ and that for sufficiently large but finite α, $\frac{1}{\alpha} \leqq \min_{j\in I} \bar{y}_j$ and $\frac{1}{\alpha} \leqq \min_{j\in J} (-\alpha g_j(\bar{x}))$. Hence for α large enough

$$\left.\begin{aligned} \theta'(\bar{y}_j) &= \frac{1}{\alpha} \quad \text{for } j \in I \\ \theta'(-\alpha g_j(\bar{x})) &= \frac{1}{\alpha} \quad \text{for } j \in J \end{aligned}\right\} \tag{10}$$

Substitution of (8) to (10) in (7) gives

$$\nabla G(\bar{y}) = I - \alpha \begin{bmatrix} (\alpha\bar{y}_I + K)^{-1}\bar{y}_I & 0 \\ 0 & \dfrac{I}{\alpha} \end{bmatrix} = \begin{bmatrix} I - (I + \bar{y}_I^{-1} \frac{K}{\alpha})^{-1} & 0 \\ 0 & 0 \end{bmatrix}$$

where $\bar{y}_I$ is a diagonal matrix with diagonal elements $\bar{y}_i$, $i \in I$. Hence for sufficiently large α, $\rho(\nabla G(\bar{y})) \leqq \dfrac{2\,\|\bar{y}_I^{-1} K\|}{\alpha}$ and again as in the extended Powell method, we have a locally convergent sequence $\{y^i\}$ with parametric root-superlinear convergence.

For concreteness we spell out algorithm 5 when the functions (8) and (9) are used.

7 Algorithm Choose $\alpha > 0$, $y^0 \in R^m$ and $x^0 \in R^n$ such that $\nabla f(x^0) + \frac{1}{2} \sum_{j=1}^{m} (\alpha g_j(x^0) + y_j^0)_+^2 \nabla g_j(x^0) = 0$. Let (x^i, y^i) determine (x^{i+1}, y^{i+1}) as follows:

(a) Determine x^{i+1} such that $\nabla f(x^{i+1}) + \frac{1}{2} \sum_{j=1}^{m} (\alpha g_j(x^{i+1}) +$

$$y_j^i)_+^2 \nabla g_j(x^{i+1}) = 0$$

or

$$f(x^{i+1}) + \frac{1}{6\alpha} \sum_{j=1}^{m} (\alpha g_j(x^{i+1}) + y_j^i)_+^3 = \min_{x\in R^n} f(x) + \frac{1}{6\alpha} \sum_{j=1}^{m} (\alpha g_j(x) + y_j^i)_+^3$$

If x^{i+1} is not unique find a closest x^{i+1} to x^i in some norm.

(b) $y_j^{i+1} = y_j^i + \alpha(\theta(|\alpha g_j(x^{i+1})+y_j^i|) + \theta(\alpha g_j(x^{i+1})) - \theta(y_j^i))$

$j = 1,\dots,m$, where θ is given by (9)

Note that in step (a) of algorithm (7) the function to be minimized is twice globally differentiable in x. This is unlike the function to be minimized in the extension of Powell's method which is

$$f(x) + \frac{1}{2\alpha}\sum_{j=1}^{m}(\alpha g_j(x)+y_j)_+^2$$ and which is not twice differentiable globally.

A possible difficulty in our proposed method is that as $\alpha \to \infty$ the function $\theta(z)$ of (9) approaches the zero function. This however is also true of Powell's function $\theta(z) = \frac{z}{2\alpha}$. Another realization of algorithm 5 which for large enough α gives (7a), the same $\nabla G(\bar{y})$ as Powell's method, is obtained from $\theta(z)$ of (9) and $\phi(z) = \alpha\theta(z)$.

In conclusion we state that the celebrated complementarity problem [7] of finding an s in R^ℓ satisfying

$$sF(s) = 0 \qquad F(s) \leqq 0 \qquad s \geqq 0 \tag{11}$$

where $F:R^\ell \to R^\ell$ is, by lemma 3 equivalent, to solving the system of equations

$$\theta(|F_i(s)+s_i|) + \theta(F_i(s)) - \theta(s_i) = 0 \quad i = 1,\dots,\ell \tag{12}$$

where θ is any strictly increasing odd function from R into R with $\theta(0) = 0$. In particular if we take $\theta(z) = z|z|$, equations (12) become

$$(F_i(s)+s_i)^2 + F_i(s)|F_i(s)| - s_i|s_i| = 0 \quad i = 1,\dots,\ell \tag{13}$$

It can be shown that the Jacobian of (13) evaluated at a solution $\bar{s}$ which satisfies the strict complementarity condition $\bar{s}_i + |F_i(\bar{s})| > 0$, $i = 1,\dots,\ell$, is nonsingular provided that the Jacobian $\nabla F(\bar{s})$ has nonsingular principal minors. This will be discussed elsewhere in more detail. Note that equations (13) are globally differentiable if $F(s)$ is globally differentiable.

ACKNOWLEDGEMENT

I am indebted to my student S. M. Chung for reading the manuscript and for a simplification in the proof of lemma 3.

REFERENCES

1. D. P. Bertsekas: "Combined primal dual and penalty methods for constrained minimization," SIAM J. Control, 13(3), August 1975.
2. D. P. Bertsekas: "Convergence rate of penalty and multiplier methods," Proceedings of 1973 IEEE Conference on Decision and Control, San Diego, California, December 1973, 260-264.
3. C. G. Broyden, J. E. Dennis & J. J. Moré: "On the local and superlinear convergence of quasi-Newton methods," J. Inst. Maths. Applics. 1973, 12, 223-245.
4. G. Debreu: "Definite and semidefinite quadratic forms," Econometrica 20, 1952, 295-300.
5. A. V. Fiacco & G. P. McCormick: "Nonlinear programming: sequential unconstrained minimization techniques", Wiley, New York, 1968.

6. R. Fletcher: "An ideal penalty function for constrained optimization," U.K.A.E.A. Report C.S.S.2, December 1973.

7. F. J. Gould & J. W. Tolle: "A unified approach to complementarity in optimization," Discrete Mathematics, 7, 1974, 225-271.

8. M. R. Hestenes: "Multiplier and gradient methods," J. Optimization Theory Appl. 4, 1969, 303-320.

9. O. L. Mangasarian: "Unconstrained lagrangians in nonlinear programming" SIAM J. Control, 13(3) August 1975.

10. J. M. Ortega & W. C. Rheinboldt: "Iterative solution of nonlinear equations in several variables," Academic Press, New York, 1970.

11. M. J. D. Powell: "A method for nonlinear constraints in minimization problems", in "Optimization", R. Fletcher, ed., Academic Press, New York, 1969, 283-298.

12. R. T. Rockafellar: "Augmented Lagrange multiplier functions and duality in nonconvex programming," SIAM J. Control 12, 1974, 268-285.

APPROXIMATION TECHNIQUES IN GRADIENT PROJECTION ALGORITHMS

H. MUKAI AND E. POLAK
Department of Electrical Engineering and Computer Sciences
and the Electronics Research Laboratory, University of California,
Berkeley, California 94720

ABSTRACT

This paper shows how one can construct gradient projection algorithms which do not require that the successively constructed points be feasible, thus overcoming a major objection to classical gradient projection algorithms.

1. INTRODUCTION

The literature on nonlinear programming contains the descriptions of a number of gradient projection algorithms [2], [4], [5], [6], [11], [12]. While all these algorithms have more or less radically differing structure, they all have one thing in common: they require that the successively constructed points be feasible. Thus, as stated, they involve an exact gradient projection or feasibility restoration sub-procedure. When the constraints are nonlinear, exact gradient projection, or restoration, is computationally not possible and even good approximations can be expensive. The literature does not contain any information as to the stability of gradient projection algorithms with respect to approximations.

In this paper we make use of the properties of an extension of Newton's method and of the approximation theory presented in [7], to develop a gradient projection algorithm with efficient approximations in the feasibility restoration phase. The approximation scheme used is not restricted to the particular method presented and can be used successfully in conjunction with various other gradient projection or reduced gradient algorithms.

2. PRELIMINARIES

The algorithm we shall present in the next section solves problems of the form

$$\min\{f(x) \mid g(x) = 0,\ h(x) \leq 0\} \tag{1}$$

where $f: \mathbb{R}^n \to \mathbb{R}^1$, is continuously differentiable and $g: \mathbb{R}^n \to \mathbb{R}^m$, $h: \mathbb{R}^n \to \mathbb{R}^\ell$ are twice continuously differentiable.

We shall denote the components of a vector by superscripts; the elements of a sequence by subscripts. Vector inequalities are to be interpreted componentwise. We shall use the Euclidean norm $\|\cdot\|$ throughout, and we shall use the notation $B(x,r) \triangleq \{x' \mid \|x-x'\| \leq r\}$, $\Omega \triangleq \{x \mid g(x) = 0,\ h(x) \leq 0\}$.

Our algorithm makes use of two special functions: $h^+, h^-: \mathbb{R}^n \to \mathbb{R}^\ell$, defined, componentwise, by $h^{+i}(x) = \max\{0, h^i(x)\}$ and $h^{-i}(x) = \min\{0, h^i(x)\}$, $i = 1 - \ell$, respectively. The restoration subalgorithm necessitates a special condition on g, h.

Assumption 1: For any $x \in \mathbb{R}^n$, the pair of Jacobian matrices $(\frac{\partial g(x)}{\partial x}, \frac{\partial h(x)}{\partial x})$ satisfies the LI condition; viz. $\{\frac{\partial g(x)}{\partial x}^T \mu + \frac{\partial h(x)}{\partial x}^T \nu = 0$, and $\nu \geq 0\}$ implies that $\mu = 0$, $\nu = 0$. #

Proposition 1: Under assumption 1, the set Ω satisfies the Kuhn-Tucher constraint qualification. (See theorem 3.3.17 [1]) #

Proposition 2: The system $\frac{\partial g(x)}{\partial x}v = a$, $\frac{\partial h(x)}{\partial x}v \leqq b$ has a solution for any $x \in \mathbb{R}^n$, $a \in \mathbb{R}^m$, $b \in \mathbb{R}^\ell$, if and only if assumption 1 holds. (See [9]) #

The algorithm in Section 3 finds Kuhn-Tucher points, which we define as follows.

Definition 1: We shall say that a point $x \in \mathbb{R}^n$ is a Kuhn-Tucker point (KTP) if $x \in \Omega$ and there exist multipliers $\mu \geqq 0$, ψ such that

$$\nabla f(x) + \frac{\partial g(x)}{\partial x}^T \psi + \frac{\partial h(x)}{\partial x}^T \mu = 0 \tag{2a}$$

and $\langle \mu, h(x) \rangle = 0$. Let

$$\mathcal{K} \overset{\Delta}{=} \{x \in \Omega \,|\, x \text{ is KTP}\}. \quad \# \tag{2b}$$

Because we shall be using only a few iterations of the restoration algorithm, at a time, we find it convenient to state that algorithm in an unconventional way. The restoration algorithm makes use of the function $p\colon \mathbb{R}^n \to \mathbb{R}^+$ defined by

$$p(x) = \frac{1}{2}\left\| \begin{matrix} g(x) \\ h^+(x) \end{matrix} \right\|^2 = \frac{1}{2}\|g(x)\|^2 + \frac{1}{2}\|h^+(x)\|^2, \tag{3}$$

and of the restoration function $A\colon \mathbb{R}^n \to \mathbb{R}^n$ defined operationally as follows:

Restoration Function A:

Note: $\alpha_1 \in (0,1/2)$, $\beta_1 \in (0,1)$ are parameters to be specified by user.

Data: $x \in \mathbb{R}^n$.

Step 1: If $p(x) = 0$, set $A(x) = x$ and stop; else go to step 2.

Step 2: Compute $v(x)$ as a solution of

$$\min\{\|v\|^2 \,|\, g(x) + \frac{\partial g(x)}{\partial x}v = 0,\ h(x) + \frac{\partial h(x)}{\partial x}v \leqq 0\}. \tag{4}$$

Step 3: Compute the smallest integer $k \geq 0$ satisfying

$$p(x + \beta_1^k v(x)) \leq (1 - 2\alpha_1\beta_1^k)\, p(x). \tag{5}$$

Step 4: Set $A(x) = x + \beta_1^k v(x)$ and stop. #

Proposition 3: The function A is well defined.

Proof: It follows from assumption 1 and proposition 2 that $v(x)$ is well defined. Next, if $p(x) \neq 0$, since $\nabla p(x) = \frac{\partial g(x)}{\partial x}^T g(x) + \frac{\partial h(x)}{\partial x}^T h^+(x)$, we have, because of (4),

$$\langle \nabla p(x), v(x) \rangle = \langle g(x), \frac{\partial g(x)}{\partial x}v(x) \rangle + \langle h^+(x), \frac{\partial h(x)}{\partial x}v(x) \rangle$$

$$\leq -\|g(x)\|^2 - \|h^+(x)\|^2 = -2p(x) < 0. \tag{5a}$$

It now follows from the mean value theorem that a finite k, satisfying (5) exists. #

The algorithm below is closely related to those of Huang [3] and Polak-Teodoru [8] and is based on Robinson's extension of Newton's method [9].

Restoration Algorithm

Data: $\beta_1 \in (0,1)$, $\alpha_1 \in (0,1/2)$, $x_o \in \mathbb{R}^n$.

Step 0: Set $i = 0$.
Step 1: Compute $A(x_i)$.
Step 2: Set $x_{i+1} = A(x_i)$, set $i = i + 1$ and go to step 1. #

In the appendix we shall prove the following.

Theorem 1: For any $x_o \in \mathbb{R}^n$ we define the sequence $Q(x_o)$ by

$$Q(x_o) = \{x_i \mid x_{i+1} = A(x_i),\ i = 0,1,2,\ldots\}. \tag{6}$$

(i) Given any x_o, either $Q(x_o)$ has no accumulation points, or $Q(x_o)$ has a unique limit point $x^* \in \Omega$ and $x_i \to x^*$ quadratically.
(ii) Given any compact set $U \subset \mathbb{R}^n$, there exist $\varepsilon > 0$, $M \in (0,\infty)$ such that for all $x_o \in S(\varepsilon) \cap U$, $Q(x_o) \subset B(x_o, M\sqrt{2p(x_o)})$, where

$$S(\varepsilon) \overset{\Delta}{=} \{x \mid \sqrt{2p(x)} \leq \varepsilon\}. \tag{7}$$

3. A GRADIENT PROJECTION ALGORITHM

A standard "conceptual" gradient projection algorithm for solving $\min\{f(x) \mid x \in \Omega\}$ (see e.g. [4]) operates as follows: given $x_i \in \Omega$, (i) compute $\nabla f(x_i)$, (ii) compute $\lambda_i \geq 0$ such that $f(\bar{d}(x_i - \lambda_i \nabla f(x_i))) = \min_{\lambda \geq 0} f(\bar{d}(x_i - \lambda \nabla f(x_i)))$, where the projection operator $\bar{d}$ is defined by $\bar{d}(x) \overset{\Delta}{=} \arg\min\{\|x - \bar{d}\|^2 \mid \bar{d} \in \Omega\}$; (iii) set $x_{i+1} = \bar{d}(x_i - \lambda_i \nabla f(x_i))$.

Clearly, in practice, λ_i and $\bar{d}(x_i - \lambda_i \nabla f(x_i))$ must be approximated. The "implementable" algorithm we shall present, incorporates efficient procedures for carrying out these approximations, in a manner compatible with preservation of convergence of the algorithm. The underlying philosophy is as follows: (i) construct a sequence $z_i \in \mathbb{R}^n$ such that $p(z_i) \searrow 0$; (ii) given a $z_i \in \mathbb{R}^n$, construct an x_k from it, by means of the restoration algorithm, such that $p(x_k) < p(z_i)$, if this is necessary to ensure an adequate cost reduction; (iii) compute approximations $y_m(x_i,\lambda)$, $m=0,1,2,\ldots$, to $\bar{d}(x_i - \lambda \nabla f(x_i))$ by means of the restoration algorithm and select λ_i to satisfy an Armijo type test, rather than cost minimization. The algorithm consists of two blocks; the master subprocedure which constructs points $\{x_k\}$ and $\{z_i\}$, and the sieve subprocedure which selects a subsequence $\{w_i\}$ of $\{z_i\}$.

Gradient Projection Algorithm

Data: $\alpha_1 \in (0,1/2)$, $\beta_1 \in (0,1)$ for restoration function A; $\alpha_2 \in (0,1)$, $\beta_2, \beta_3, \beta_4 \in (0,1)$, $\gamma_1, \gamma_2 > 0$, $\varepsilon_0 > 0$, $c_o > 0$, $\lambda_s \in (0,1]$, $\lambda_{\min} \in (0, \lambda_s/\varepsilon_o)$; $z_0 \in \mathbb{R}^n$.

I. Master Subprocedure.

Step 0: Set $i = 0$, $\varepsilon = \varepsilon_0$, $j = 0$.

Comment: $j = 0$ initializes the sieve subprocedure.
Step 1: Set $k = 0$, $x_0 = z_i$.
Step 2: If $x_k \in S(\varepsilon)$, go to step 4; else, go to step 3.

Step 3: Compute $x_{k+1} = A(x_k)$, set $k = k + 1$ and go to step 2.

Step 4: Compute

$$d(x_k) \triangleq \arg\min_{d \in \mathbb{R}^n} \{\|\nabla f(x_k) + d\|^2 \mid \frac{\partial g(x_k)}{\partial x} d = 0,\ h^-(x_k) + \frac{\partial h(x_k)}{\partial x} d \leqq 0\} \quad (8)$$

and $\phi(x_k) \triangleq \|\nabla f(x_k) + d(x_k)\|^2$. (9)

Step 5: If $\|d(x_k)\| \geq \gamma_1 \varepsilon$, go to step 6; else, set $\varepsilon = \beta_2 \varepsilon$ and go to step 2.

Step 6: Set $\ell = 0$, $\lambda = \lambda_s$.

Step 7: If $\lambda \geqq \lambda_{min} \varepsilon$, go to step 8; else, set $\varepsilon = \beta_2 \varepsilon$ and go to step 2.

Step 8: Set $m = 0$, $y_o = x_k + \lambda d(x_k)$.

Step 9: If $y_m \in S(\varepsilon)$, go to step 11; else, go to step 10.

Step 10: Compute $y_{m+1} = A(y_m)$, set $m = m + 1$ and go to step 9.

Step 11: If $f(y_m) - f(x_k) \leq \alpha_2 \lambda \langle \nabla f(x_k), d(x_k) \rangle$, go to step 12; else, set $\ell = \ell + 1$, $\lambda = \beta_3 \lambda$ and go to step 7.

Step 12: If $f(y_m) - f(x_k) \leq -\gamma_2 \varepsilon$, go to step 13, else set $\varepsilon = \beta_2 \varepsilon$ and go to step 2.

Step 13: Set $z_{i+1} = y_m$, (set $\varepsilon_{i+1} = \varepsilon$), set $i = i + 1$ and go to step 14.

Comment: ε_i is defined for use in proving lemma 7 only.

II. Sieve Subprocedure

Step 14: Compute $d(z_i)$.

Step 15: If $\|d(z_i)\| \leq c_j$, set $w_j = z_i$, $c_{j+1} = \beta_4 c_j$, $j = j + 1$, and go to step 1; else, go to step 1.

We shall prove in the appendix the following

Lemma 1: The function $d\colon \mathbb{R}^n \to \mathbb{R}^n$ defined by (8) is continuous.

Lemma 2: A point $x^* \in \mathbb{R}^n$ is a Kuhn-Tucker point if and only if $p(x^*) = 0$ and $d(x^*) = 0$.

Proof: ⇒ If x^* is a Kuhn-Tucker point, then $p(x^*) = 0$ and $d = 0$ satisfies the Kuhn-Tucker conditions for (8). Since the Kuhn-Tucker conditions are both necessary and sufficient for (8), $d(x^*) = 0$.

⇐ Suppose $p(x^*) = 0$ and $d(x^*) = 0$, then applying the Kuhn-Tucker conditions to (8), we immediately obtain that x^* is a Kuhn-Tucker point. #

Lemma 3: For any $x \in \mathbb{R}^n$, (i) $\|d(x)\| \leq 2\|\nabla f(x)\|$, and (ii) $\langle \nabla f(x), d(x) \rangle \leq -\frac{1}{2}\|d(x)\|^2$.

Proof: Since θ is feasible for the min problem in (8), $\phi(x) \leq \|\nabla f(x)\|^2$. Hence,

$$\|d(x)\| \leq \|d(x) + \nabla f(x)\| + \|f(x)\| = \sqrt{\phi(x)} + \|\nabla f(x)\| \leq 2\|\nabla f(x)\|. \quad (10)$$

Hence (ii) follows from

$$\|\nabla f(x)\|^2 + \|d(x)\|^2 + 2\langle \nabla f(x), d(x) \rangle = \phi(x) \leq \|\nabla f(x)\|^2. \quad (11)$$

In what follows we depend on the assumption below, which will generally be satisfied.

Assumption 2: For any $x \in \mathbb{R}^n$, the sequence $Q(x)$ (see (6)) has a unique limit point.

Lemma 4: Given any compact subset $\Lambda \subset \Omega$, such that $\Lambda \cap \mathcal{K} = \phi$, there exist reals $\varepsilon_\Lambda > 0, \rho_\Lambda > 0$ and an integer $\ell_\Lambda \geq 0$ such that for any $x \in \bigcup_{z\in\Lambda} B(z,\rho_\Lambda)$ and $y \in Q(x+\beta_3^{\ell_\Lambda}\lambda_s d(x)) \cap S(\varepsilon_\Lambda)$,

$$f(y) - f(x) \leq \alpha_2\beta_3^{\ell_\Lambda}\lambda_s \langle \nabla f(x), d(x)\rangle. \tag{12}$$

Proof: Let $\Lambda \subset \Omega$, with $\Lambda \cap \mathcal{K} = \phi$, be compact and let U be a compact neighborhood of Λ. Then, because of theorem 1, there exist $\varepsilon_\Lambda > 0$, $M \in (0,\infty)$ such that $Q(x) \subset B(x, M\sqrt{2\rho(x)})$, $\forall x \in U \cap S(\varepsilon_\Lambda)$. Also, because of the continuity of $d(\cdot)$ (lemma 1) and because $U \cap S(\varepsilon_\Lambda)$ is a neighborhood of a compact Λ, there exist a $\rho_1 > 0$ and a $\lambda_1 \in (0,1]$ such that $(x+ d(x)) \in U \in S(\varepsilon_\Lambda)$, $\forall x \in \bigcup_{z\in\Lambda} B(z,\rho_1)$, $\forall \lambda \in [0,\lambda_1]$. Now the functions $f, \frac{\partial g}{\partial x}, \frac{\partial h}{\partial x}$ are continuously differentiable, and hence there exists a Lipschitz constant $L > 0$ as used below.

Let $x \in \bigcup_{z\in\Lambda} B(z,\rho_1)$, $\lambda \in (0,\lambda_1]$ and $y \in Q(x+\lambda d(x)) \cap S(\varepsilon_\Lambda)$. Then, making use of lemma 3,

$$f(y)-f(x)-\lambda\alpha_2 \langle \nabla f(x), d(x)\rangle =$$

$$[f(y)-f(x+\lambda d(x))] + f(x+\lambda d(x))-f(x)-\lambda\alpha_2 \langle \nabla f(x),d(x)\rangle \leq LM\sqrt{2p(x+\lambda d(x))}$$

$$+ \lambda \sup_{s\in[0,1]} \|\nabla f(x+s\lambda d(x))-\nabla f(x)\| \, \|d(x)\| - \lambda(1-\alpha_2)\, \|d(x)\|^2/2 \tag{13}$$

Let $\gamma\colon \mathbb{R}^n \times \mathbb{R}^1 \to \mathbb{R}^1$ be defined by

$$\gamma(x,\lambda) \triangleq LM\sqrt{2p(x+\lambda d(x))} + \lambda \sup_{s\in[0,1]} \|\nabla f(x+s\lambda d(x))-\nabla f(x)\| \, \|d(x)\| - \lambda(1-\alpha_2)\|d(x)\|^2/2\,. \tag{14}$$

Now, for any $z \in \Lambda$, $\lambda \in [0,\lambda_1]$.

$$\|g(z+\lambda d(z))\| \leq L\,\lambda^2\, \|d(z)\|^2, \tag{15}$$

$$\|h^+(z+\lambda d(z))\| \leq L\,\lambda^2\, \|d(z)\|^2 \tag{16}$$

and hence

$$\sqrt{2p(z+\lambda d(z))} \leq \sqrt{2}\,L\,\lambda^2\, \|d(z)\|^2 \tag{17}$$

Hence, for all $z \in \lambda$, $\lambda \in [0,\lambda_1]$,

$$\gamma(z,\lambda) \leq \lambda[\sqrt{2}\,L^2M\lambda\|d(z)\| + \sup_{s\in[0,1]} \|\nabla f(z+s\lambda d(z)) - \nabla f(z)\| - \lambda(1-\alpha_2)\, \|d(z)\|/2]\, \|d(z)\| \tag{18}$$

and there exists a $\lambda_2 \in (0,\lambda_1]$ such that $\forall z \in \Lambda$, $\lambda \in [0,\lambda_2]$, $\gamma(z,\lambda) \leq -\lambda(1-\alpha_2)d^*/4$ where $d^* = \min_{z\in\Lambda} \|d(z)\|^2 > 0$. It now follows from the uniform continuity of γ on $\bigcup_{z\in\Lambda} B(z,\rho_1) \times [0,\lambda_2]$ that there exists a $\rho_\Lambda \in (0,\rho_1]$ and an $\ell_\Lambda \geq 0$, with $\lambda_s\beta_3^{\ell_\Lambda} \in (0,\lambda_2]$ such that

$$\gamma(x,\beta_3^{\ell_\Lambda}\lambda_s) \leq -\beta_3^{\ell_\Lambda}\lambda_s(1-\alpha_2)\, d^*/8, \ \forall x \in \bigcup_{z\in\Lambda} B(z,\rho_\Lambda)$$

which completes our proof. #

Our algorithm contains a number of loops. We now show that given any loop, the algorithm either leaves it after a finite number of cycles or cycles indefinitely in it while constructing an $x^* \in \mathcal{K}$. The following result is obvious.

Lemma 5: The gradient projection algorithm cannot cycle indefinitely in the loops defined by steps 2-3, 7-11, 9-10. #

Lemma 6: Suppose that the gradient projection algorithm cycles indefinitely in one of the loops defined by steps 2-5, or 2-11, or 2-12, then either it jams up at an $x_k \in \mathcal{K}$ or it constructs an infinite sequence $\{x_k\}_{k=0}^{\infty}$ such that $x_k \to x^* \in \mathcal{K}$.

Proof: (i) Loop defined by steps 2-5. Suppose the algorithm cycles indefinitely in this loop, constructing a sequence $\{x_k\}$. If $\{x_k\}_{k=0}^{\nu}$ is finite, then there exists a j_o such that $x_\nu \in S(\varepsilon_o\beta_2^j)$ and $\|d(x_\nu)\| \leq \gamma_1 \varepsilon_o\beta_2^j$ for all integers $j \geq j_o$ and hence $p(x_\nu) = 0$, $d(x_\nu) = 0$, i.e. $x_\nu \in \mathcal{K}$. If $\{x_k\}_{k=0}^{\infty}$ is infinite, then, by assumption 2 and theorem 1, $x_k \to x^* \in \Omega$, and there exists a subsequence $\{x_{k_j}\}_{j=0}^{\infty}$ such that $\|d(x_{k_j})\| < \gamma_1\varepsilon_o\beta_2^j$, $j = j_o, j_o + 1,\ldots$. Since $d(\cdot)$ is continuous, $d(x^*) = 0$, i.e. $x^* \in \mathcal{K}$.

(ii) Loop defined by steps 2-11. Suppose the algorithm cycles indefinitely in this loop (but not in 2-5), without reaching step 12 while constructing a sequence $\{x_k\}$. If $\{x_k\}_{k=0}^{\nu}$ is finite, then there exists a j_o such that for all $j \geq j_o$ (a) $x_\nu \in S(\varepsilon_o\beta_2^j)$ and (b) for a subsequence $\{y_{m_j}\}_{j=0}^{\infty}$ of $\{y_m\}$

$$f(y_{m_j}) - f(x_\nu) > \alpha_2\beta_3^j\lambda_s\langle\nabla f(x_\nu),\, d(x_\nu)\rangle \tag{19}$$

with $y_{m_j} \in S(\varepsilon_o\beta_2^j) \cap Q(x_\nu+\beta_3^j\lambda_s d(x_\nu))$. From (z), $x_\nu \in \Omega$ and it now follows from (19) and lemma 4 that $x_\nu \in \mathcal{K}$.

Next, suppose that the sequence $\{x_k\}$ is infinite. Then, because of assumption 2 and theorem 1, $x_k \to x^* \in \Omega$. It now follows by a similar argument to the one above that $x^* \in \mathcal{K}$, for otherwise there would be a contradiction of lemma 4.

(iii) Loop defined by steps 2-12. Suppose the algorithm cycles indefinitely in this loop (but not in 2-5 or 2-11), without reaching step 13, while constructing a sequence $\{x_k\}$. If $\{x_k\}_{k=0}^{\nu}$ is finite, then, arguing as above, we find that $x_\nu \in \Omega$. Suppose that $x_\nu \notin \mathcal{K}$. Then by lemma 4, there exists an ℓ_ν and an $\varepsilon_\nu > 0$ such that for all $y_m \in Q(x_\nu+\beta_3^{\ell_\nu}\lambda_s d(x_\nu)) \cap S(\varepsilon_\nu)$, $f(y_m)-f(x_\nu) \leq \alpha_2\beta_3^{\ell_\nu}\lambda_s\langle\nabla f(x_\nu),d(x_\nu)\rangle < 0$ and hence the test in step 12 will be passed after a finite number of cycles. Thus, we must have $x_\nu \in \mathcal{K}$.

Now suppose that $\{x_k\}_{k=0}^{\infty}$ is infinite. Then, as before, $x_k \to x^* \in \Omega$, and if $x^* \notin \mathcal{K}$, then again from lemma 4, we conclude that there must exist a k' such that the test in step 12 is passed by $x_{k'}$, after a finite number of cycles, passing to step 13. But this contradicts our assumption that $\{x_k\}$ is infinite. Thus we must have $x^* \in \mathcal{K}$. #

Lemma 7: Suppose that the gradient projection algorithm constructs an infinite sequence $\{z_i\}_{i=0}^{\infty}$. If $\{z_i\}$ has accumulation points, then all the accumulation points of $\{z_i\}_{i=0}^{\infty}$ are in Ω and the accompanying sequence $\{\varepsilon_i\}_{i=0}^{\infty}$ converges to zero.

Proof: Since $\{\varepsilon_i\}_{i=0}^{\infty}$ is monotone decreasing and $\varepsilon_i > 0$, we must have $\varepsilon_i \to \varepsilon^* \geq 0$. Suppose $\varepsilon^* > 0$, then there exists an i' such that $\varepsilon_i = \varepsilon^* > 0$ for all $i \geq i'$. From the test in step 9, it follows that $z_i \in S(\varepsilon^*)$ for all $i \geq i'$ and hence for any $i \geq i'$ step 12 is reached with $k = 0$ and the test is passed, i.e. for all $i \geq i'$,

$$f(z_{i+1}) - f(z_i) \leq - \gamma_2 \, \varepsilon^*. \tag{20}$$

But this implies that $f(z_{i_j}) \to \infty$ as $i_j \to \infty$ on a bounded subsequence $\{z_{i_j}\}$, which contradicts the continuity of $f(\cdot)$. Hence $\varepsilon_i \to 0$ as $i \to \infty$.

Next, since $z_i \in S(\varepsilon_i)$, $i = 0,1,2,\ldots$, and $\varepsilon_i \to 0$, and accumulation point z^* of $\{z_i\}$ must be in $S(0) = \Omega$. #

Lemma 8: Suppose that the gradient projection algorithm constructs a bounded infinite sequence $\{z_i\}$. Then $\{z_i\}$ has at least one accumulation point $z^* \in \mathcal{K}$.

Proof: Let Λ be the set of accumulation points of $\{z_i\}$ and suppose that $\Lambda \cap \mathcal{K} = \phi$. Since Λ is compact and $\varepsilon_i \to 0$ by lemma 7, and since $z_i \in S(\varepsilon_i)$, it follows from lemma 4 that there exists an i' and ℓ_Λ such that for all $i \geq i'$,

$$f(z_{i+1}) - f(x_{k(i)}) \leq \alpha_2 \beta_3^{\ell_\Lambda} \lambda_s \langle \nabla f(x_{k(i)}), d(x_{k(i)}) \rangle \leq \frac{1}{2} \alpha_2 \beta_3^{\ell_\Lambda} \lambda_s \lambda^* \tag{21}$$

where $\lambda^* = \max\limits_{x \in \Lambda} \langle \nabla f(x), d(x) \rangle < 0$ and $x_{k(i)}$ is the point used to construct z_{i+1}. Next, by Theorem 1, there exists an M and an $i'' \geq i'$ such that, since $x_{k(i)} \in Q(z_i)$ and $z_i \in S(\varepsilon_i)$, $\|x_{k(i)} - z_i\| \leq M \varepsilon_i$. Hence, making use of (21), there exists a Lipschitz constant L such that

$$\begin{aligned} f(z_{i+1}) - f(z_i) &\leq \frac{1}{2} \alpha_2 \beta_3^{\ell_\Lambda} \lambda_s \lambda^* - [f(z_i) - f(x_{k(i)})] \\ &\leq \frac{1}{2} \alpha_2 \beta_3^{\ell_\Lambda} \lambda_s \lambda^* + L\, M\, \varepsilon_i, \quad i \geq i''. \end{aligned} \tag{22}$$

But (22) implies that $f(z_i) \to \infty$, which is impossible. Hence $\Lambda \cap \mathcal{K} \neq \phi$. #

Lemma 9: Suppose that the gradient projection algorithm constructs a bounded infinite sequence $\{z_i\}$. Then it also constructs an infinite sequence $\{w_i\}$, all of whose accumulation points are in $\mathcal{K}$.

Proof: This lemma follows directly from lemmas 1, 2 and 8.

We can now summarize our findings as follows.

Theorem 2: (i) If the gradient projection algorithm jams up at a point x_ν, then $x_\nu \in \mathcal{K}$. (ii) If the gradient projection algorithm constructs either an infinite sequence $\{x_k\}$ or an infinite sequence $\{w_j\}$ then any accumulation point of these sequences is in $\mathcal{K}$. (iii) If neither (i) nor (ii) takes place, then the gradient projection algorithm constructs an infinite unbounded sequence $\{z_i\}$.

4. Conclusion

We have shown in this paper that a major objection to classical gradient projection algorithms, arising from their requirement that successive points be feasible, can be overcome. This has been accomplished by utilizing some of the properties of an extended Newton type method and the approximation theory described in [7]. The results presented are of obvious theoretical interest, and, in addition, they should prove to be of computational value, as well.

References

[1] M. D. Canon, C. D. Cullum Jr. and E. Polak, Theory of Optimal Control and Mathematical Programming, McGraw-Hill, 1970.

[2] A. A. Goldstein, "Convex Programming in Hilbert Space," Bull. of AMS, Vol. 70, No. 5, pp. 709-710, 1964.

[3] T. J. Huang, "Algorithms for Solving Systems of Equations and Inequalities with Applications in Nonlinear Programming," Computer Sciences Tech. Rept. #191, University of Wisconsin, Sept. 1973.

[4] E. S. Levitin and B. T. Polyak, "Constrained Minimization Methods," USSR Comp. Math. and Math. Physics, Vol. 6, No. 5, pp. 1-50, 1966.

[5] D. G. Luenberger, "The Gradient Projection Method Along Geodesics," Management Science, Vol. 18, No. 11, pp. 620-631, 1972.

[6] G. P. McCormick and R. A. Tapia, "The Gradient Projection Method Under Mild Differentiability Conditions," SIAM J. Control, Vol. 10, No. 1, pp. 93-98, 1972.

[7] E. Polak, Computational Methods in Optimization, Academic Press, 1971.

[8] E. Polak and I. Teodoru, "Newton Derived Methods for Nonlinear Equations and Inequalities," ERL Memo No. ERL-M449, Electronics Research Lab., University of California, Berkeley, 1974.

[9] S. M. Robinson, "Extension of Newton's Method to Mixed Systems of Nonlinear Equations and Inequalities," Tech. Sum. Rept. No. 1161, Mathematical Research Center, University of Wisconsin, 1971.

[10] S. M. Robinson, "Bounds for Error in the Solution Set of a Perturbed Linear Program," Linear Algebra and Its Applications, Vol. 6, pp. 69-81, 1973.

[11] J. B. Rosen, "The Gradient Projection Method for Nonlinear Programming. Part I. Linear Constraints," J. of SIAM, Vol. 8, pp. 181-217, 1960.

[12] J. B. Rosen, "The Gradient Projection Method for Nonlinear Programming. Part II. Nonlinear Constraints," J. of SIAM, Vol. 9, No. 4, pp. 514-532, 1961.

Appendix

In the next two proofs we shall need the following result: For any $x \in \mathbb{R}^n$, we define

$$\mu(x) \triangleq \max\left\{\min\left\{\|v\| \,\middle|\, \frac{\partial g(x)}{\partial x}v = a,\ \frac{\partial h(x)}{\partial x}v \leqq b\right\} \,\middle|\, \left\|\begin{matrix}a\\ b\end{matrix}\right\| \le 1\right\}. \tag{A1}$$

Because of assumption 1, it follows from [9] and [10] that $\mu(\cdot)$ is well defined and upper semicontinuous.

Proof of theorem 1

First, since any v satisfying $g(x) + \frac{\partial g(x)}{\partial x}v = 0$, $h^+(x) + \frac{\partial h(x)}{\partial x}v \leqq 0$ also satisfies the constraints in (4), we must have

$$\|v(x)\| \le \mu(x) \left\|\begin{matrix}g(x)\\ h^+(x)\end{matrix}\right\| = \mu(x)\sqrt{2p(x)}\,. \tag{A2}$$

Then, making use of (5a), the continuity of $p(\cdot)$, the fact that $\nabla p(x) = 0$ if and only if $p(x) = 0$ and theorem 1.3.10 in [7], we conclude that for any $x_o \in \mathbb{R}^n$, any accumulation point x^* of $Q(x_o)$ satisfies $p(x^*) = 0$, i.e., $x^* \in \Omega$.

Next, if $Q(x_o)$ has accumulation points, then $p(x_i) \to 0$, $v(x_i) \to 0$ as $i \to \infty$. It now follows from [9] that there exists an i' such that the sequence $\{\xi_j\}_{j=0}^{\infty}$ defined by

$$\xi_{j+1} = \xi_j + v(\xi_j),\ j = 0,1,2,\ldots,\ \xi_o = x_{i'} \tag{A3}$$

converges to an $x^* \in \Omega$. Furthermore, there exists an $M \in (0,\infty)$ such that

$$\|\xi_{j+1} - \xi_j\| \le M\|\xi_j - \xi_{j-1}\|^2,\ j=1,2,\ldots\ . \tag{A4}$$

Now, from (4) we have that $\|g(\xi_j)\| \le \left\|\frac{\partial g(\xi_j)}{\partial x}\right\| \|v(\xi_j)\|$, and, $h^k(\xi_j) \le -\left(\frac{\partial h(\xi_j)}{\partial x}v(\xi_j)\right)^k$, $k=1,2,\ldots,\ell$. Hence $\|h^+(\xi_j)\| \le \left\|\frac{\partial h(\xi_j)}{\partial x}\right\| \|v(\xi_j)\|$. Therefore

$$p(\xi_j) \le M' \|v(\xi_j)\|^2, \tag{A5}$$

where $M' = \frac{1}{2}\sup_j \left(\left\|\frac{\partial g(\xi_j)}{\partial x}\right\|^2 + \left\|\frac{\partial h(\xi_j)}{\partial x}\right\|^2\right)$. Combining (A2), (A3) and (A4) we get

$$p(\xi_{j+1}) \le M' \|v(\xi_{j+1})\|^2 \le M' M^2 \|v(\xi_j)\|^4 \le 4\mu(\xi_j)^2 M' M^2 p(\xi_j)^2 \tag{A6}$$

Since $\mu(\cdot)$ is u.s.c. and $\xi_j \to x^*$, there exists an $M^* \in (0,\infty)$ such that

$$p(\xi_{j+1}) \le (M^* p(\xi_j))\, p(\xi_j),\ j=0,1,2,\ldots \tag{A7}$$

It can be shown that i' can be chosen sufficiently large to ensure that $M^*p(\xi_j) \le (1-2\alpha_1)$ for $j = 0,1,2,\ldots$. Hence, in the restoration algorithm, we must have $x_{i'+j} = \xi_j$ for $j = 0,1,2,\ldots$. Thus, part (i) of theorem 1 is true. Part (ii) of theorem 1 now follows directly from [9]. #

Proof Lemma 1

Suppose that $d(\cdot)$ is not continuous at x^*, then, because of lemma 3, there exists a sequence $\{x_i\}_{i=0}^{\infty}$ such that $x_i \to x^*$, but $d(x_i) \to d^* \neq d(x^*)$. Since $\frac{\partial g(\cdot)}{\partial x}$, $\frac{\partial h(\cdot)}{\partial x}$ and $h^-(\cdot)$ are continuous, it follows that $\frac{\partial g(x^*)}{\partial x} d^* = 0$, $h^-(x^*) + \frac{\partial h(x^*)}{\partial x} d^* \leq 0$ and hence that $\phi(x^*) < \|\nabla f(x^*) + d^*\|^2 \triangleq \phi^*$. Now there exists a sequence $\{d_i\}_{i=0}^{\infty}$ such that d_i satisfies the constraints in (8) and for $i = 0,1,2,\ldots$.

$$\|d_i - d(x^*)\| \leq \mu(x_i) \left\| \begin{matrix} \left(\frac{\partial g(x^*)}{\partial x} - \frac{\partial g(x_i)}{\partial x}\right) d(x^*) \\ h^-(x^*) - h^-(x_i) + \left(\frac{\partial h(x^*)}{\partial x} - \frac{\partial h(x_i)}{\partial x}\right) d(x^*) \end{matrix} \right\| \tag{A8}$$

Hence $d_i \to d(x^*)$, $\phi(x_i) \leq \|\nabla f(x_i) + d_i\|^2$, and $\|\nabla f(x_i) + d_i\|^2 \to \phi(x^*)$. But by assumption $\phi(x_i) \to \phi^* > \phi(x^*)$ and hence we have a contradiction. #

Research sponsored by the Joint Services Electronics Program, Contract F44620-71,C-0087 and the Army Research Office, Contract DAHC04-73-C-0025.

DISCRETE OPTIMAL LINEAR SMOOTHING FOR SYSTEMS WITH UNCERTAIN OBSERVATIONS

R. A. Monzingo
Hughes Aircraft Company
Fullerton, California

ABSTRACT

The smoothing filter and smoothing error covariance matrix equations are developed for discrete linear systems whose observations may contain noise alone, where only the probability of occurrence of such cases is known to the estimator. An example of such a system arises in trajectory tracking where the signal is first detected and then the estimator processes the signal for tracking purposes. The results apply to any detection decision process, however, since any such decision is associated with a false alarm probability, which is the probability that the detected signal contains only noise. The present results together with the earlier work of Nahi on prediction and filtering give a complete treatment of the discrete linear estimation problem for systems characterized by uncertain observations. These results of course reduce to well-known formulations for the classical estimation problem in the case where the observation is always assumed to contain the signal to be estimated.

I. INTRODUCTION

The modern literature dealing with classical estimation theory [1], [2] adopted the viewpoint of orthogonal projection to develop the fundamental equations of optimal linear filtering and prediction for both discrete and continuous linear systems. The results derived in this manner were subsequently derived using a variety of other approaches including least squares [3], maximum likelihood [4], [5], linear regression [6], dynamic programming [7], stochastic approximation [8], and the Bayesian approach [9]. Likewise, the literature dealing with the classical smoothing problem adopted a number of different viewpoints including the calculus of variations [10], maximum likelihood [5], and orthogonal projection [11]. More recently the classical smoothing problem results were derived by regarding the optimum smoother as a combination of two optimum filters [12].

The fundamental equations of optimal linear filtering and prediction have recently been extended to the case where the system is characterized by uncertain observations [13]. Such systems are important whenever a decision concerning the presence of a valid signal is made, as with trajectory tracking where a decision concerning the existence of a target must be made. The result of each detection decision, however, has an associated probability of false alarm. The extension of the optimum smoothing equations in this paper to include systems characterized by uncertain observations then provides a complete treatment of the estimation problem for systems of this type. The development of the fundamental smoothing equations here adopts the two optimum filter approach, since this permits the previously derived filtering equations to be used directly and results in a particularly simple formulation that is easily manipulated to yield different forms of the resulting smoothing solution.

II. PROBLEM STATEMENT

Consider the discrete linear system described by

$$x(k+1) = \Phi(k+1,k)\ x(k) + G(k)\ u(k) \tag{1}$$

$$z(k+1) = \gamma(k+1)\ H(k+1)\ x(k+1) + v(k+1) \tag{2}$$

where x is a n-component state vector; z is a m-component output vector; Φ is a real n x n state transition matrix; and H is a real m x n matrix. In addition, u and v are independent, zero mean, Gaussian white sequences for which

$$E\{u(j)\ u'(k)\} = Q(k)\ \delta_{jk} \tag{3}$$

$$E\{v(j)\ v'(k)\} = R(k)\ \delta_{jk} \tag{4}$$

where δ_{jk} is the Kronecker delta and where Q is a real n x n positive semidefinite matrix and R is a real m x m positive definite matrix. The initial state x(0) is assumed to be a zero mean Gaussian random vector which is independent of u and v for all k, and for which

$$E\{x(0)\ x'(0)\} = \mathrm{Cov}\ (0) \tag{5}$$

where Cov (0) is a real n x n positive semidefinite matrix that represents the apriori statistics of the initial condition.

The quantity $\gamma(k)$ is a random scalar taking on the values of 0 and 1 with

$$\Pr\{\gamma(k) = 1\} \triangleq p(k) \tag{6}$$

$$\Pr\{\gamma(k) = 0\} \triangleq 1 - p(k) \tag{7}$$

Note that 1 - p(k) is just the probability of occurrence of a false alarm. From (6) and (7) it follows that the random scalar $\gamma(k)$ has

$$E\{\gamma(k)\} = p(k) \tag{8}$$

$$E\{\gamma(j)\ \gamma(k)\} = p(j)\ p(k),\ j \neq k \tag{9}$$

and

$$E\{\gamma^2(k)\} = p(k) \tag{10}$$

The problem of concern in this paper is to obtain an estimate of x(k) based on measurements up to and including the one at time n (where $k < N$), denoted by $\hat{x}(k/N)$, and the associated error covariance matrix P(k/N). A particularly simple way of approaching this problem is to regard the optimum linear smoother as a combination of two optimum linear filters, one of which operates forward over the data and the other of which runs backward over the data interval [12].

III. THE TWO FILTER APPROACH TO LINEAR SMOOTHING

The equations for the optimal discrete linear smoother expressed as a combination of a forward-running and a backward-running filter are given by [12]

$$\hat{x}(k/N) = P(k/N)[E^{-1}(k/k)\ \hat{x}(k/k) + P^{-1}(k/k+1)\ \hat{x}_b(k/k+1) \tag{11}$$

$$P(k/N) = [E^{-1}(k/k) + P^{-1}(k/k+1)]^{-1} \tag{12}$$

where $\hat{x}(k/k)$ and $E(k/k)$ are the (filtered) state estimate and associated error covariance matrix corresponding to the forward-running filter and $\hat{x}_b(k/k+1)$ and $P(k/k+1)$ are the (predicted) state estimate and associated error covariance matrix corresponding to the backward-running filter. It is shown in Appendix I that (11) and (12) can be rearranged to yield the more familiar forms associated with the fixed-interval smoothed estimate:

$$\hat{x}(k/N) = \hat{x}(k/k) + J(k)\,[\hat{x}(k+1/N) - \hat{x}(k+1/k)] \tag{13}$$

$$P(k/N) = E(k/k) + J(k)\,[P(k+1/N) - E(k+1/k)]\,J^T(k) \tag{14}$$

where

$$J(k) = E(k/k)\,\Phi^T(k+1,k)\,E^{-1}(k+1/k) \tag{15}$$

with the boundary condition $P(N/N) = E(N/N)$.

A problem arises associated with applying equation (12) directly for two reasons: 1) $P(k/k+1)$ is not finite when $k = N$. 2) Equation (12) requires three inversions of matrices which are of the dimension of the state. The problem of the infinite $P(k/k+1)$ can be avoided by using its inverse to obtain an expression which does not involve any infinite quantities and which only requires one inversion of a matrix of the dimension of the state. The resulting equation is derived in Appendix II and is given by

$$P(k/N) = [I - W(k)\,P^{-1}(k/k+1)]\,E(k/k)\,[I - W(k)\,P^{-1}(k/k+1)]^T + W(k)\,P^{-1}(k/k+1)\,W^T(k) \tag{16}$$

where

$$W(k) = E(k/k)(I + E(k/k)\,P^{-1}(k/k+1))^{-T} \tag{17}$$

and

$$P^{-1}(k/k+1) = 0 \text{ when } k = N.$$

Calculation of the backward filter state estimate directly involves the backward filter covariance matrix $P(k/k+1)$ and the terminal condition $\hat{x}_b(N/N+1)$, both of which are undesirable. These problems may be avoided by introducing

$$W_b(k/k+1) = P^{-1}(k/k+1)\,\hat{x}_b(k/k+1) \tag{18}$$

It is shown in Appendix III that the smoothed state estimate can then be expressed as

$$\hat{x}(k/N) = [I + E(k/k)\,P^{-1}(k/k+1)]^{-1}\,\hat{x}(k/k) + P(k/N)\,W_b(k/k+1) \tag{19}$$

where

$$P(k/N)\,W_b(k/k+1) = 0 \text{ when } k = N.$$

The recursion relation that $W_b(k/k+1)$ must satisfy is developed in the next section.

IV. RESULTS FOR SYSTEMS WITH UNCERTAIN OBSERVATIONS

The smoothing solutions presented in Section III depend on the state estimates and associated covariance matrices of forward and backward-running filters that are appropriate for systems characterized by uncertain observation processes. Nahi [13] derived the appropriate expressions for a forward-running filter for which the filtered state estimate is given by

$$\hat{x}(k/k) = \hat{x}(k/k-1) + p(k)\ E(k/k-1)\ H^T(k)\ [R(k) + p^2(k)\ H(k)\ E(k/k-1)\ H^T(k)$$
$$+ p(k)(1 - p(k))\ H(k)\ S(k)\ H^T(k)]^{-1}(z(k) - p(k)\ H(k)\ x(k/k-1)) \quad (20)$$

and the associated filtered error covariance matrix is

$$E(k/k) = E(k/k-1) - p^2(k)\ E(k/k-1)\ H^T(k)\ [R(k) + p^2(k)\ H(k)\ E(k/k-1)\ H^T(k)$$
$$+ p(k)\ (1 - p(k))\ H(k)\ S(k)\ H^T(k)]^{-1}\ H(k)\ E(k/k-1) \quad (21)$$

or

$$E(k/k) = [E^{-1}(k/k-1) + p^2(k)\ H^T(k)\ (R(k) + p(k)(1 - p(k))\ H(k)\ S(k)$$
$$\times H^T(k))^{-1}\ H(k)]^{-1} \quad (22)$$

where

$S(k)$ is defined to be $E\{x(k/k-1)\ x^T(k/k-1)\}$ and satisfies

$$S(k+1) = \Phi(k+1,k)\ S(k)\ \Phi^T(k+1,k) + G(k)\ Q(k)\ G^T(k) \quad (23)$$

with the boundary condition $S(0) = \text{Cov}\ (0)$ (24)
where Cov (0) is the covariance of the initial condition $x(0)$ and represents the apriori statistics.

The corresponding predicted quantities are then found using

$$\hat{x}(k+1/k) = \Phi(k+1,k)\ \hat{x}(k/k) \quad (25)$$

$$E(k+1/k) = \Phi(k+1,k)\ E(k/k)\ \Phi^T(k+1,k) + G(k)\ Q(k)\ G^T(k) \quad (26)$$

The quantities $\hat{x}(k/k)$, $\hat{x}(k+1,k)$, and $E(k/k)$, $E(k+1/k)$ provided by (20), (21) and (25), (26) together with (13), (14), and (15) provide the fixed interval smoothing solution for systems characterized by undertain observation processes. It now remains to develop a computationally more desirable solution by considering the expressions that obtain for a backward-running filter.

For a backward-running filter the appropriate expression for the filtered state estimate is given by

$$\hat{x}_b(k/k) = \hat{x}_b(k/k+1) + p(k)\ P(k/k+1)\ H^T(k)\ [R(k) + p^2(k)\ H(k)\ P(k/k+1)\ H^T(k)$$
$$+ p(k)(1 - p(k))\ H(k)\ V(k)\ H^T(k)]^{-1} \times (z(k) - p(k)\ H(k)\ \hat{x}_b(k/k+1))$$
$$(27)$$

and the inverse of the associated filtered error covariance matrix is

$$P^{-1}(k/k) = P^{-1}(k/k+1) + p^2(k)\ H^T(k)[R(k) + p(k)(1 - p(k))\ H(k) \\ \times V(k)\ H^T(k)]^{-1}\ H(k) \tag{28}$$

where V(k) is defined to be $E\{x(k/k+1)\ x^T(k/k+1)\}$ and satisfies the relation

$$V(k) = \Phi^{-1}(k+1,k)\ [V(k+1) + G(k)\ Q(k)\ G^T(k)]\ \Phi^{-T}(k+1,k) \tag{29}$$

The quantity V(N) represents the apriori uncertainty associated with the state x(N) so that V(N) = S(N) is the appropriate boundary condition to be imposed on (29). Since the variable $P^{-1}(k/k+1)$ appears in (16) and (17), a recursive relation for this quantity is desired. From equation (40) of Appendix I, it follows that

$$\Phi^{-T}(k,k-1)\ P^{-1}(k-1/k)\ \Phi^{-1}(k,k-1) = P^{-1}(k/k) \\ - P^{-1}(k/k)\ G(k-1)[G^T(k-1)\ P^{-1}(k/k)\ G(k-1) + Q^{-1}(k-1)]^{-1} \\ \times G(k-1)\ P^{-1}(k/k) \tag{30}$$

$$= [I - J(k)\ G^T(k-1)]\ P^{-1}(k/k) \tag{31}$$

where $J(k) \triangleq P^{-1}(k/k)\ G(k-1)[G^T(k-1)\ P^{-1}(k/k)\ G(k-1) + Q^{-1}(k-1)]^{-1}$ (32)
Equation (31) may then be expanded into a symmetric form to yield

$$P^{-1}(k-1/k) = \Phi^T(k,k-1)[(I - J(k)\ G^T(k-1)\ P^{-1}(k/k)(I - J(k)\ G^T(k-1))^T \\ + J(k)\ Q^{-1}(k-1)\ J^T(k)]\ \Phi(k,k-1) \tag{33}$$

and

$$\hat{x}_b(k/k+1) = \Phi^{-1}(k+1,k)\ \hat{x}_b(k+1/k+1) \tag{34}$$

Equation (33) in conjunction with (16), (17), (28), and (29) then provides a convenient computational means of obtaining the smoothed error covariance matrix in which the forward filter covariance matrix and the inverse of the backward filter covariance matrix are required.

It now remains to obtain a recursive relation for $W_b(k/k+1)$. From (31) it follows that

$$P^{-1}(k/k+1)\ \Phi^{-1}(k+1,k) = \Phi^T(k+1,k)\ [I - J(k+1)\ G^T(k)]\ P^{-1}(k+1/k+1) \tag{35}$$

Postmultiplying (35) by $x_b(k+1/k+1)$ and using the fact that

$$W_b(k/k+1) = P^{-1}(k/k+1)\ \Phi^{-1}(k+1/k)\ \hat{x}_b(k+1/k+1) \tag{36}$$

then yields

$$W_b(k/k+1) = \Phi^T(k+1,k)[I - J(k+1)\ G^T(k)]\ W_b(k+1/k+1) \tag{37}$$

Equations (37) and (19) then yield a computationally desirable solution for the smoothed state estimate of systems having uncertain observations.

V. CONCLUSIONS

This paper has presented a derivation of the equations of discrete optimal linear smoothing appropriate for discrete linear systems having uncertain observations by regarding the smoothing process as the combination of a forward-running and a backward-running filter. The fixed-interval smoothing problem was treated here, but the results obtained here can easily be manipulated to obtain the appropriate expressions for both the fixed-point and the fixed-lag smoothing problems. Analogous derivations can also be applied to obtain corresponding results for linear continuous systems. These results, when combined with Nahi's results [13] on prediction and filtering give a complete treatment of the estimation problem for linear systems characterized by uncertain observations.

APPENDIX I

It is desired to show that equations (11) and (12) are fully equivalent to equations (13) and (14), respectively. Starting with (12), it follows that

$$P(k/N)^{-1} = E^{-1}(k/k) + P^{-1}(k/k+1) \tag{38}$$

Since the backward-running filter is concerned with a process whose transitions are governed by

$$x(k) = \Phi^{-1}(k+1,k)\,[x(k+1) - G(k)\,u(k)] \tag{39}$$

it follows that

$$P(k/k+1) = \Phi^{-1}(k+1,k)\,[P(k+1/k+1) + G(k)\,Q(k)\,G^T(k)]\,\Phi^{-T}(k+1,k) \tag{40}$$

but since

$$P^{-1}(k+1/k+1) = P^{-1}(k+1/N) - E^{-1}(k+1/k) \tag{41}$$

Equation (38) can be expressed as

$$\begin{aligned} P^{-1}(k/N) = E^{-1}(k/k) + \Phi^T(k+1,k)[&(P^{-1}(k+1/N) - E^{-1}(k+1/k))^{-1} \\ &+ G(k)\,Q(k)\,G^T(k)]^{-1}\,\Phi(k+1,k) \end{aligned} \tag{42}$$

Applying the matrix inversion lemma to (42) and recalling that

$$E(k+1/k) = \Phi(k+1,k)\,E(k/k)\,\Phi^T(k+1,k) + G(k)\,Q(k)\,G^T(k) \tag{43}$$

yields

$$\begin{aligned} P(k/N) = E(k/k) - E(k/k)\,\Phi^T(k+1,k)\,[&E(k+1/k) + (P^{-1}(k+1/N) \\ &- E^{-1}(k+1/k))^{-1}]\,\Phi(k+1,k)\,E(k/k) \end{aligned} \tag{44}$$

Again applying the matrix inversion lemma to the quantity within the inner brackets in (43) and recalling that J(k) is given by (10) yields

$$P(k/N) = E(k/k) + J(k)[P(k+1/N) - E(k+1/k)]\,J(k) \tag{45}$$

which shows that equation (12) is equivalent to equation (14).

It now remains to show that equation (13) is equivalent to equation (11). Substitute (14), (15), and (40) into (11) and use the fact that

$$\hat{x}_b(k/k+1) = \Phi^{-1}(k+1,k)\ \hat{x}_b(k+1/k+1) \tag{46}$$

to obtain

$$\begin{aligned}
\hat{x}(k/N) = {} & \hat{x}(k/k) + E(k/k)\ \Phi^T(k+1,k)[P(k+1/k+1) + G(k)\ Q(k)\ G^T(k)]^{-1} \\
& \times \hat{x}_b(k+1/k+1) + J(k)\ P(k+1/N)(E^{-1}(k+1/k)\ \Phi(k+1/k)\ E(k/k)) \\
& \times E^{-1}(k/k)\ \hat{x}(k/k) \\
+ {} & J(k)\ P(k+1/N)(E^{-1}(k+1/k)\ \Phi(k+1/k)\ E(k/k))\ \Phi^T(k+1/k)\ (P(k+1/k+1) \\
& + G(k)\ Q(k)\ G^T(k))^{-1}\ \hat{x}_b(k+1/k+1) \\
- {} & J(k)\ E(k+1/k)(E^{-1}(k+1/k)\ \Phi(k+1/k)\ E(k/k))\ E^{-1}(k/k)\ \hat{x}(k/k) \\
- {} & J(k)\ E(k+1/k)(E^{-1}(k+1/k)\ \Phi(k+1/k)\ E(k/k))\ \Phi^T(k+1/k)(P(k+1/k+1) \\
& + G(k)\ Q(k)\ G^T(k))^{-1}\ \hat{x}_b(k+1/k+1)
\end{aligned} \tag{47}$$

Expanding the above expression by carrying out multiplication and recognizing that:

$$\Phi(k+1/k)\ \hat{x}(k/k) = \hat{x}(k+1/k) \tag{48}$$

$$\Phi(k+1/k)\ E(k/k)\ \Phi^T(k+1/k) = E(k+1/k) - G(k)\ Q(k)\ G^T(k) \tag{49}$$

$$J(k)\ E(k+1/k) = E(k/k)\ \Phi^T(k+1/k) \tag{50}$$

the resulting expression can be simplified by cancellation and combining of like terms to yield

$$\begin{aligned}
\hat{x}(k/N) = {} & \hat{x}(k/k) - J(k)\ \hat{x}(k+1/k) + J(k)\ P(k+1/N)\ E^{-1}(k+1/k)\ \hat{x}(k+1/k) \\
& + J(k)\ P(k+1/N) \\
& \times \underbrace{[I + P^{-1}(k+1/k+1)\ G(k)\ Q(k)\ G^T(k)]\left(P(k+1/k+1) + G(k)Q(k)G^T(k)\right)^{-1}}_{P^{-1}(k+1/k+1)} \\
& \times \hat{x}_b(k+1/k+1)
\end{aligned} \tag{51}$$

or

$$\begin{aligned}
\hat{x}(k/N) = {} & \hat{x}(k/k) - J(k)\ \hat{x}(k+1/k) \\
& + J(k)\ \underbrace{P(k+1/N)[E^{-1}(k+1/k)\ x(k+1/k) + P^{-1}(k+1/k+1)\ \hat{x}_b(k+1/k+1)]}_{\hat{x}(k+1/N)}
\end{aligned} \tag{52}$$

so that

$$\hat{x}(k/N) = \hat{x}(k/k) + J(k)\ [\hat{x}(k+1/N) - \hat{x}(k+1/k)] \tag{53}$$

which demonstrates that equation (11) is equivalent to equation (13).

APPENDIX II

We wish to show that equations (16) and (17) provide an alternative means of computing the smoothed error covariance matrix. Beginning with equation (12), it follows that

$$P^{-1}(k/N) = E^{-1}(k/k) + P^{-1}(k/k+1) \tag{54}$$

Multiplying (54) by E(k/k) and inverting the result yields

$$P(k/N) = (I + E(k/k)\ P^{-1}(k/k+1)\ E(k/k) \tag{55}$$

$$= E(k/k) - E(k/k)\ P^{-1}(k/k+1) \times (I + E(k/k)\ P^{-1}(k/k+1))\ E(k/k) \tag{56}$$

or

$$P(k/N) = (I - W(k)\ P^{-1}(k/k+1))\ E(k/k)(I - W(k)\ P^{-1}(k/k+1))^T + W(k)\ P^{-1}(k/k+1)\ W^T(k) \tag{57}$$

where

$$W(k) \triangleq E(k/k)\ (I + E(k/k)\ P^{-1}(k/k+1))^{-T} \tag{58}$$

which is the desired result. The symmetric form represented by equation (57) is a more desirable computational form than equation (56), since it is less sensitive to arithmetic truncation.

APPENDIX III

We wish to show that the smoothed state estimate can be expressed as given by equation (19). Substituting equation (13) into equation (11) yields

$$\hat{x}(k/N) = P(k/N)\ E^{-1}(k/k+1)\ \hat{x}(k/k) + P(k/N)\ W_b(k/k+1) \tag{59}$$

$$= [E^{-1}(k/k) + P^{-1}(k/k+1)]^{-1} E^{-1}(k/k)\ x(k/k) + P(k/N) \times W_b(k/k+1) \tag{60}$$

$$= E^{-1}(k/k)(I + E(k/k)\ P^{-1}(k/k+1))]^{-1}\ E^{-1}(k/k)\ \hat{x}(k/k) + P(k/N)\ W_b(k/k+1) \tag{61}$$

or

$$\hat{x}(k/N) = (I + E(k/k)\ P^{-1}(k/k+1))^{-1}\ \hat{x}(k/k) + P(k/N)\ W_b(k/k+1) \tag{62}$$

which is the desired result.

REFERENCES

[1] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," Trans. ASME, Series D, J. Basic Engrg, Vol. 82D (1960), pp 35-45.

[2] R. E. Kalman and R. S. Bucy, "New Results in Linear Filtering and Prediction Theory, Trans. ASME, Series D, J. Basic Engrg, Vol. 83D (1961), pp 95-107.

[3] Y. C. Ho, "The Method of Least Squares and Optimal Filtering Theory," RM-3329-PR, The Rand Corporation, Santa Monica, California.

[4] R. H. Battin, Astronautical Guidance, McGraw-Hill, New York, 1964 pp 303-340.

[5] H. E. Rauch, F. Tung, and C. T. Striebel, "Maximum Likelihood Estimates of Linear Dynamic Systems," AIAA J., Vol. 3 (1965), pp 1445-1450.

[6] S. L. Fagin, "Recursive Linear Regression Theory, Optimal Filter Theory, and Error Analysis of Optimal Systems," 1964 IEEE Convention Record, Part I, New York, 1964, pp 216-240.

[7] H. Cox, "On the Estimation of State Variables and Parameters for Noisy Dynamic Systems," IEEE Trans. Automatic Control, Vol. AC-9 (1964) pp 5-12.

[8] Y. C. Ho, "On the Stochastic Approximation Method and Optimal Filtering Theory," J. Math. Anal. Appl., Vol. 6 (1962), pp 152-154.

[9] Y. C. Ho and R.C.K.Lee, "A Bayesian Approach to Problems in Stochastic Estimation and Control," Joint Automatic Control Conference Proceedings, Stanford, 1964.

[10] A. E. Bryson and M. Frazier, "Smoothing for Linear and Nonlinear Dynamic Systems," TDR-63-119, Aeronautical Systems Division, Wright-Patterson AFB, Dayton, 1963.

[11] J. S. Meditch, "Orthogonal Projection and Discrete Optimal Linear Smoothing," J. SIAM Control, Vol. 5, No. 1 (1967), pp 74-89.

[12] D. C. Fraser and J. E. Potter, "The Optimum Linear Smoother as a Combination of Two Optimum Linear Filters," IEEE Trans. Automatic Control, Vol. AC-14, No. 4 (1969), pp 387-390.

[13] N. E. Nahi, "Optimal Recursive Estimation with Uncertain Observation," IEEE Trans. Information Theory, Vol. IT-15, No. 4 (1969), pp 457-462.

PRECISION ON-BOARD ATTITUDE REFERENCE USING THE GENERALIZED INVERSE

T. T. MC ELROY and R. P. IWENS
TRW Systems, Redondo Beach, California

ABSTRACT

A sequential estimation algorithm is presented for spacecraft attitude determination using the generalized inverse (pseudoinverse). Relative to the Kalman filter, computational requirements are thereby significantly reduced. The sensors used are a three-axis strapdown gyro package and strapdown star trackers. Digital simulation in conjunction with a truth model is employed to evaluate performance, indicating three-axis accuracy better than 30 arc seconds, 3σ.

INTRODUCTION

This paper describes a simple method for precise attitude determination for a slowly rotating spacecraft (e.g. scanning or three-axis stabilized earth pointing) using gyros and strapdown star trackers. The algorithm has obvious extensions to inertially pointing spacecraft, provided at least two stars can be observed simultaneously by the tracker(s). Attitude estimates are at all times obtained from the integrated gyro outputs. Because of gyro bias and random drift, and uncertainty in the inertial attitude, attitude updates from star data are required. The software data flow of the proposed on-board attitude reference system is shown in Figure 1. Gyro bias updates are determined on the ground with state propagation and control functions implemented in an on-board processor. Since one star sighting (yielding azimuth and elevation) does not provide sufficient information for a complete three-axis attitude update, a sequential estimation scheme is required. Kalman filtering techniques, extended to nonlinear systems, could be applied as has been reported at various times in the technical literature [1, 2, 3]. In order to reduce the computational burden on the on-board computer, it is desirable to look for a simple sequential estimation scheme. Since most of the computations required in the Kalman filtering algorithm are incurred in propagating the error covariance matrix, it is natural to look for a sequential filter that does not require the availability of the covariance matrix. However, at the same time, this filter must still be sensitive to changes in the system and its environment. The latter consideration rules out a fixed gain Kalman filter. The sequential filtering algorithm proposed here makes use of the generalized matrix inverse for computing an attitude update each time a star is sighted.

SYSTEM DESCRIPTION

The generalized inverse sequential filter algorithm will now be described in more detail. In essence, the difficulty of insufficient information in one star sighting is overcome by generating an attitude update that on one hand is consistent with the star observation made, and on the other hand remains as close as possible to the a priori attitude estimate available from the integrated gyro outputs. The generalized inverse precisely accomplishes this. The price one pays for not propagating and using the error covariance matrix in the update computations is mainly twofold: (1) filter convergence is slowed down, and (2) updates to gyro bias estimates cannot be obtained from the on-board computations, but must be provided from ground-based intelligence at infrequent intervals (on the order of several days). The benefit one obtains when the covariance computations are eliminated is that computer storage requirements are reduced approximately 50% with a sizeable reduction in execution time. For many missions the advantages of a simpler, and computationally less demanding on-board algorithm outweigh the disadvantages.

Spacecraft kinematics are described by Euler symmetric parameters, i.e. the quaternion vector $\underline{q}$. The attitude estimate is propagated with a state transition technique which utilizes integrated rate gyro information. The measurement equations model the star tracker outputs as a function of true attitude. Stars are oriented relative to an inertial coordinate system by the star right ascension α and star declination δ. The projection of the i'th star vector onto the sensor coordinate frame is given by

$$\underline{u}_S^i = A_{SB} A_{BI}(\underline{q}) \, \underline{u}_I^i(\alpha^i, \delta^i) \tag{1}$$

where the functional dependence has been indicated and A_{SB} is a known 3 x 3 matrix describing the orientation of the tracker in the spacecraft. The nominal tracker readings are given by

$$\underline{z} = \underline{F}(\underline{\beta}^j, \underline{u}_S^i) \tag{2}$$

where $\underline{\beta}^j$ is a vector containing the misalignment of the j'th tracker and $\underline{u}_S^i$ is defined above. The fourth component of the quaternion vector can be eliminated by imposing the quaternion constraint equation. The first three components of the quaternion vector are defined by the vector $\underline{q}^o$. The attitude update equation becomes then

$$\Delta\hat{\underline{q}}^o = H^\dagger \Delta\underline{z} \tag{3}$$

where $H^\dagger$ is the pseudoinverse of the measurement matrix H. The measurement matrix H is a 2 x 3 matrix defined by

$$H \triangleq \frac{\partial \underline{z}}{\partial \underline{q}^o} \tag{4}$$

and the measurement residual $\Delta\underline{z}$ is the difference between the actual and estimated star tracker readings.

A sufficient condition for algorithm convergence in the mean is that the state transition matrix Φ of the (linearized) system satisfy $\Phi\Phi^T = I$. This is a rather strong condition but it is satisfied when quaternions are used for the spacecraft kinematic description. Convergence in the mean-square sense can be investigated by a covariance analysis. The matrix covariance equation is given by

$$P_{k+1} = \Psi P_k \Psi^T + H_{k+1}^\dagger Q_{k+1}^* H_{k+1}^{\dagger T} \tag{5}$$

where

$$\Psi = (I - H_{k+1}^\dagger H_{k+1})\Phi_k$$

and Q_{k+1}^* is the covariance of a combination of gyro and star sensor noise.

Insight into the estimation scheme can be obtained graphically when the problem is artificially reduced to estimating two fixed parameters from two, not simultaneously available, linearly independent, one-dimensional measurements. The state and measurement perturbation equations for this artificial system would be given by

$$\begin{aligned} \Delta\underline{\theta}_k &= \Delta\underline{\theta}_{k-1} + \Delta\underline{b}\,T &&: \text{ STATE EQUATION} \\ \Delta z_k &= H_k\,\Delta\underline{\theta}_k &&: \text{ MEASUREMENT EQUATION} \end{aligned} \qquad (6)$$

where by definition

$$\Delta\underline{\theta}_k = \underline{\theta}_k - \hat{\underline{\theta}}_k$$

$$\Delta\underline{b} = \underline{b} - \hat{\underline{b}}$$

The θ's are analogous to spacecraft attitude, b's are analogous to gyro bias, and z is analogous to star measurements. The processing of alternate measurements is illustrated in Figure 2, showing convergence to a stable limit cycle with offset error due to $\Delta\underline{b}$.

SIMULATION RESULTS

Simulation of on-board reference performance under conditions of noise, biases, and computation algorithm models is investigated in this section. Representative results for scan and point mode operation is illustrated in Figures 3 and 4 respectively. One hour of real time performance is shown in both cases.

In the scan mode, attitude updates are made whenever a guide star enters into the sensor field of view (e.g. 8° x 8°). As shown in Figure 3, three guide stars per scan revolution are sufficient for algorithm convergence and maintaining an accurate attitude reference. After elimination of the a priori estimation error, system performance is indicated by short term gyro performance. As shown in Figure 3, this "steady" state is reached after one scan revolution.

In the inertial point mode, if star separation is limited (e.g. only one active tracker), simultaneous processing of two stars is required. The pseudo-inverse of the now (4 x 3) measurement matrix is equivalent to the classical least squares estimator. Good three axis attitude performance is obtained as shown by the performance data of Figure 4. A slight increase in computing requirements is involved due to the increased dimensionality of the generalized inverse of the measurement matrix and attendant update computational complexity. This increase is negligible compared to the performance advantages gained. The simulation data shows that the attitude estimate errors are maintained below 10 arc seconds after correction of the initial error.

SUMMARY

A spacecraft attitude estimation algorithm is presented which utilizes the generalized matrix inverse. Simulation results indicate that good three axis attitude is obtained in both the scan and point modes. In the point mode simultaneous tracking of two separate stars is a requirement.

REFERENCES

1. N. F. Toda, J. L. Heiss, and F. H. Schlee, "SPARS: The System, Algorithms, and Test Results", Proceedings of the Symposium on Spacecraft Attitude Determination, Vol. 1, pp. 361-370, October 1969.
2. J. E. Lesinski, "Attitude Determination Performance Potential for a Yaw-spin Satellite, Part One", Proceedings of the Symposium of Spacecraft Attitude Determination, Vol. 1, pp. 397-413, October 1969.
3. R. P. Iwens and R. L. Farrenkopf, "Performance Evaluation of a Precision Attitude Determination System (PADS)", AIAA Guidance, Control and Flight Mechanics Conference, Paper No. 71-1964, August 1971.

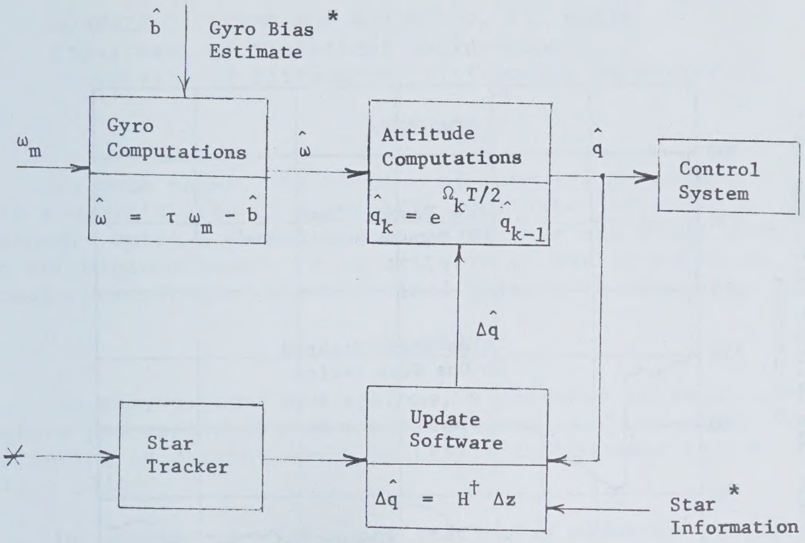

$\hat{b}$
Gyro Bias *
Estimate
ω_m
Gyro
Computations
$\hat{\omega} = \tau\, \omega_m - \hat{b}$
$\hat{\omega}$
Attitude
Computations
$\hat{q}_k = e^{\Omega_k T/2}\, \hat{q}_{k-1}$
$\hat{q}$
Control
System
$\Delta\hat{q}$
Star
Tracker
Update
Software
$\Delta\hat{q} = H^{\dagger}\, \Delta z$
Star *
Information

Figure 1. Functional Block Diagram of On-Board Attitude Reference System (*Ground Generated Information)

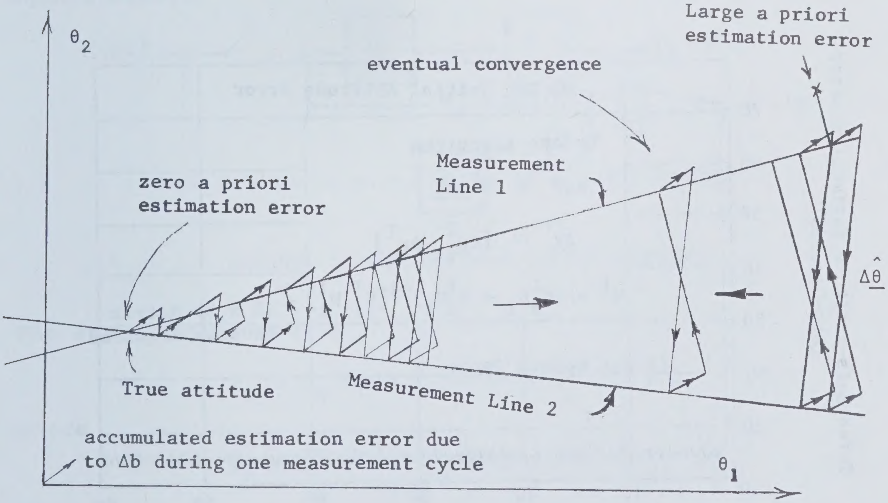

θ_2
Large a priori
estimation error
eventual convergence
Measurement
Line 1
zero a priori
estimation error
$\Delta\hat{\underline{\theta}}$
True attitude
Measurement Line 2
accumulated estimation error due
to Δb during one measurement cycle
θ_1

Figure 2. Sequential Filter Convergence Geometry

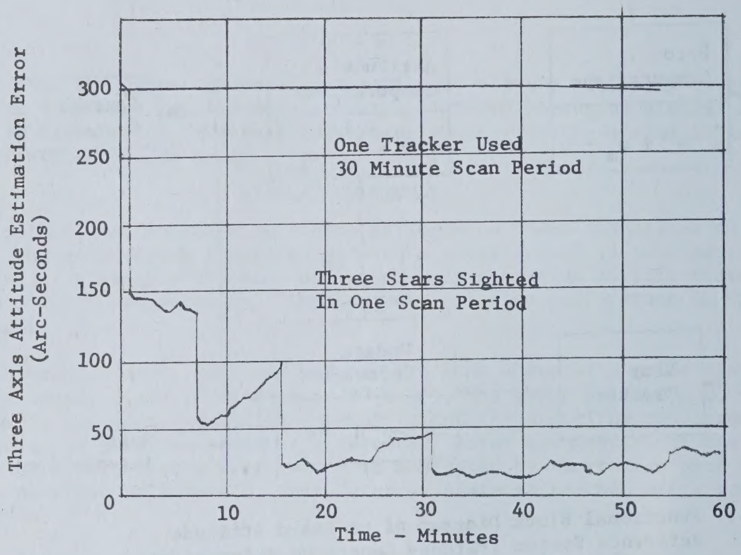


Figure 3. Scan Mode Simulation Results

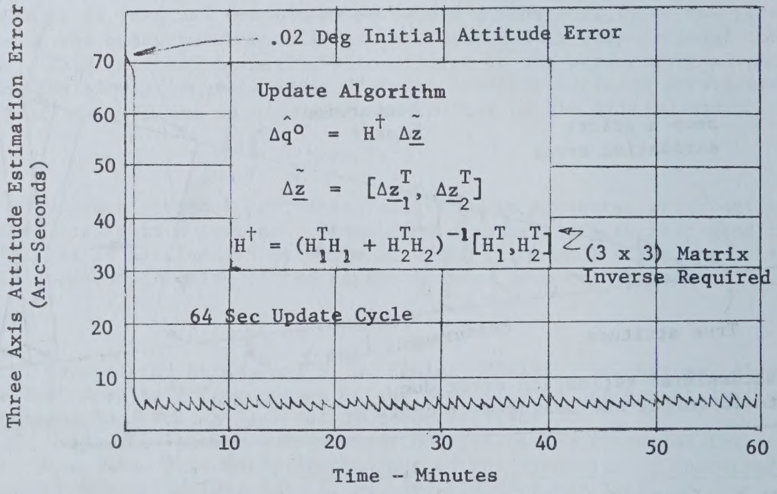


Figure 4. Point Mode Simulation Results When Processing Two Star Measurements, z_1 and z_2, Simultaneously

A TIME DOMAIN ANALYSIS OF OPTIMAL FILTERING IN MULTIRATE SYSTEMS

JOHNNIE C. LIANG and MICHAEL G. STRINTZIS
Department of Electrical Engineering
University of Pittsburgh, Pittsburgh, Pennsylvania

ABSTRACT

In this paper, the problem of matching an arbitrary digital signals's rate to a specific slow rate processor is examined. Optimal prefilter and postfilter are found according to the minimum square error criterion. Easily realizable time domain description of the optimal filters is obtained.

SUMMARY

In many digital PAM systems, a decrease in sampling rate before processing is sometimes required, and consequently prefiltering is desirable and postfiltering becomes necessary [1] - [3].

We consider a wide-sense stationary sequence $\{X(nT), n = 1, 2,3,4....\}$ with known autocorrelation function $R_X(\cdot)$ which is to be processed by a processor G operating at the slower rate of one sample per MT seconds with M a positive integer. It is desired to find a prefilter H so as to minimize the mean square error.

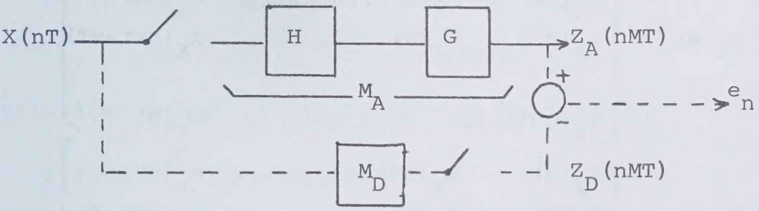


FIGURE 1 Actual and Ideal Systems

The error is defined as

$$e_n = Z_A(nMT) - Z_D(nMT)$$

where

$$Z_A(nMT) = \sum_{K=0}^{N} M_A(KMT)X(nMT-KMT)$$

$$Z_D(nMT) = \sum_{K=0}^{N} M_D(KT)X(nMT-KT)$$

M_D is the ideal system while M_A is the combined system of prefilter and the slow rate processor.

$$M_A(KMT) = \sum_{K=0}^{N} H(pMT)G(KMT-pMT)$$

All above filters were assumed, for simplicity, to be of equal impulse response length N. The optimal prefilter is determined by selecting H(0) to H(NMT) so as to minimize the mean square error $E\{e^2\}$. Following lengthy but essentially straightforward calculations one obtains:

$$\underline{H} = \underline{G}_0^{-1} \cdot \underline{R}_1^{-1} \cdot \underline{R}_2^{T} \cdot \underline{M}_D$$

where $\underline{H}$ and $\underline{M}_D$ are (N + 1) -demensional vectors:

$$\underline{H} = (H(0)H(MT)H(2MT) \ldots H(NMT))^T$$

$$\underline{M}_D = (M_D(0)M_D(T)M_D(2T) \ldots M_D(NT))^T$$

and the matrices $\underline{G}_0$, $\underline{R}_1$, $\underline{R}_2$ are:

$$\underline{G}_0 = \begin{bmatrix} G(0) & 0 & 0 & 0 \cdots\cdots & 0 \\ G(MT) & G(0) & 0 & 0 \cdots\cdots & 0 \\ \vdots & & & & \\ G(NMT) \cdots & \cdots & \cdots & \cdots\cdots & G(0) \end{bmatrix}$$

$$\underline{R}_1 = \begin{bmatrix} R_X(0) & R_X(MT) \cdots\cdots & R_X(NMT) \\ R_X(MT) & R_X(0) \cdots\cdots & R_X(NMT-MT) \\ \vdots & & \\ R_X(NMT) \cdots & \cdots\cdots & R_X(0) \end{bmatrix}$$

$$\underline{R}_2 = \begin{bmatrix} R_X(0) & R_X(MT) \cdots\cdots & R_X(NMT) \\ R_X(T) & R_X(MT-T) \cdots\cdots & R_X(NMT-T) \\ \vdots & & \\ R_X(NT) & R_X(MT-NT) \cdots\cdots & R_X(NMT-NT) \end{bmatrix}$$

The minimum mean square error is easily found to be

$$e_{min} = \underline{M}_D^T \, [\underline{R}_0 - \underline{R}_2 \underline{R}_1^{-1} \underline{R}_2^T] \underline{M}_D$$

where $\underline{R}_0$ is the following (N + 1)X(N + 1) matrix

$$\underline{R}_0 = \begin{bmatrix} R_X(0) & R_X(T) \ldots\ldots\ldots\ldots & R_X(NT) \\ R_X(T) & R_X(0) \ldots\ldots\ldots\ldots & R_X(NT-T) \\ \vdots & & \\ R_X(NT) & R_X(NT-T) \ldots\ldots\ldots & R_X(0) \end{bmatrix}$$

As expected, the error reduces to zero if M = 1, i.e., if no rate reduction is necessary for processing.

Determination of the optimal postfilter is considered next. The task of the postfilter P in Figure 2 is to restore the processed sequence to its original rate of one sample per T seconds.

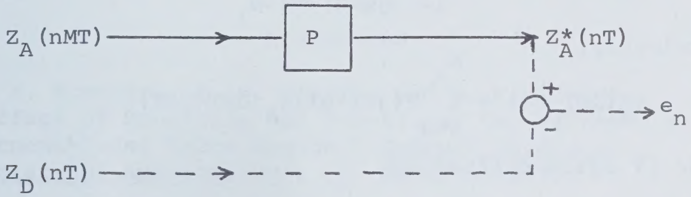


FIGURE 2 Postfilter Design

It will be helpful to define a new sequence W(nT)[4]:

$$W(nT) = Z_A[(n/M)MT] \qquad n = 0, M, 2M.....$$

$$= 0 \qquad \text{otherwise}$$

The postfilter output is easily seen to be given by

$$Z_A^*(nT) = \sum_{K=0}^{N} P(KT)W(nT-KT)$$

$$= \sum_{s=n-N}^{N} P(nT-sT)W(sT)$$

or, equivalently,

$$Z_A^*(nT) = \sum_{i=[(n-N)/M]}^{[n/M]} P(nT-iMT)Z_A(iMT)$$

where [•] denotes the integer part of its argument. Without loss of generality, we choose N to N_1M, i.e. an integer multiple of M. Now, for n = QM and Q a positive integer:

$$Z_A^*(QMT) = \sum_{i=Q-N_1}^{Q} P(QMT-iMT)Z_A(iMT)$$

$$= \sum_{j=0}^{N_1} P(jMT) Z_A(QMT-jMT)$$

Since the values of the output of P at the original sample times must be the same as the original sample values, it is clear that

$$P(0) = 1, \quad P(jMT) = 0, \quad 1 \leq j \leq N_1$$

if n is not an integer multiple of N, say, n = QM + B, where B is an integer such that $1 \leq B \leq (M-1)$, then

$$Z_A^*(QMT + BT) = \sum_{i=[(QM+B)/M]-N_1}^{[(QM+B)/M]} P(QMT+BT-iMT) Z_A(iMT)$$

or equivalently,

$$Z_A^*(QMT+BT) = \sum_{j=0}^{N_1} P(jMT+BT) Z_A(QMT*jMT)$$

the error is again defined as

$$e_n = Z_A^*(nT) - Z_D(nT)$$

On minimizing the mean square error $E\{e_n^2\}$ we obtain

$$\underline{P}_B = \underline{U}^{-1} \quad \underline{S}_B$$

$$e_{\min(n=QM+B)} = \underline{M}_D^T \underline{R}_0 \underline{M}_D - \underline{S}_B^T \underline{U}^{-1} \underline{S}_B$$

where

$$\underline{P}_B = (P(BT) \; P(MT+BT) \; P(2MT+BT) \ldots . P(N_1MT+BT))^T$$

characterizes the optimal postfilter. The matrices and vectors $\underline{Y}$, $\underline{U}$, $\underline{V}_B$ and $\underline{S}_B$ are defined below.

$$\underline{Y}(K-J) = \begin{pmatrix} R_X(-(K-J)MT) & R_X(MT-(K-J)MT) \ldots R_X(N_1MT-(K-J)MT) \\ R_X(-MT-(K-J)MT) & \ldots\ldots\ldots\ldots R_X(N_1MT-MT-(K-J)MT) \\ \vdots & \\ R_X(-N_1MT-(K-J)MT) & \ldots\ldots\ldots\ldots R_X(-(K-J)MT) \end{pmatrix}$$

$$\underline{U} = \sum_{K=0}^{N} \sum_{J=0}^{N} M_A(KMT) M_A(JMT) \underline{Y}(K-J)$$

$$\underline{V}_B = (R_X(BT-nT+iMT) \; R_X(MT+BT-nT+iMT) \ldots R_X(N_1MT+BT-nT+iMT))^T$$

$$\underline{S}_B = \sum_{n=0}^{N} \sum_{i=0}^{N} M_A(iMT) M_D(nT) \underline{V}_B(n,i)$$

The filter configuration is given in Figure 3.

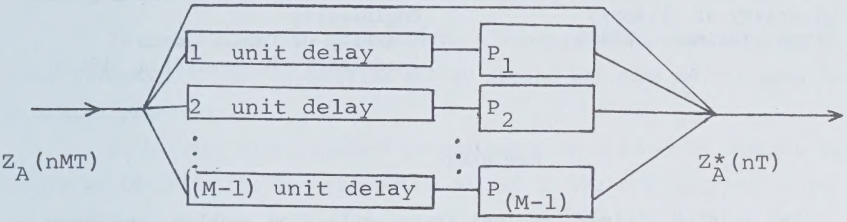


FIGURE 3 Postfilter Configuration

REFERENCES

[1] A. E. Roseberg, R. W. Schafer and L. R. Rabiner, "Effect of Smoothing and Quantizing the Parameters of Forment-Coded Voice Speech," Journal of Acoustic Society of America, Vol. 50, December 1971.

[2] R. W. Schafer and L. R. Rabiner, "Design and Simulation of a Speech Analysis-Synthesis System Based on Short-Time Fourier Analysis," IEEE Transactions on Audio and Electroacoustics, June 1973.

[3] D. J. Goodman, "The Application of Delta-Modulation to Analog-to-PCM Encoding," Bell System Technical Journal, Vol. 48, February 1969.

[4] R. W. Schafer and L. R. Rabiner, "A Digital Signal Processing Approach to Interpolation," IEEE Proceedings, June 1973.

MINIMAX TERMINAL STATE ESTIMATION UNDER EXPONENTIATED GENERALIZED QUADRATIC LOSS*

C. PADILLA
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois 61801

M. MINTZ
Moore School of Electrical Engineer
Engineering
University of Pennsylvania
Philadelphia, Pennsylvania 19174

ABSTRACT

Two related minimax terminal state estimation problems are posed for a linear plant and exponentiated generalized quadratic loss functions. Sufficient conditions are developed in each case to insure that a Kalman filter will provide a minimax estimate for the terminal state of the plant under each loss function considered. In particular, for each case the existence of a bounded solution for a specific Riccati equation over the time interval $[0,T]$ guarantees that a Kalman filter will provide a minimax estimate.

1. INTRODUCTION

This paper considers two related minimax terminal state estimation problems. These estimation problems can serve as mathematical models for a radar tracking problem in which one is attempting to track a blind target which is under the control of an intelligent adversary and has been programmed to maneuver evasively. The minimax formulation reflects a game theoretic situation in which we are confronted with an evader who is faced with a trade off between a gain due to our estimation error versus a loss due to the use of energy for evasive maneuvering. In this context our model could represent a blind incoming missile which has been preprogrammed to maneuver evasively.

The target is modeled as a known linear dynamic system with an unknown forcing function which is assumed to be under the control of our intelligent adversary. The estimates are minimax with respect to the class of exponentiated generalized quadratic loss functions. This class of loss functions provides us with means of rationally modeling the conflicting

---

*This work was supported in part by the Joint Services Electronics Program (U. S. Army, U. S. Navy, and U. S. Air Force) under Contract DAAB-07-72-C-0259, and in part by the U. S. Air Force under Grant AFOSR-73-2570.

objectives of the evader, namely the trade off between estimation error and maneuvering energy.

2. STATEMENTS OF TERMINAL STATE ESTIMATION PROBLEMS I AND II

In order to properly delineate a stochastic dynamic game, the prior information and information gained during the play of the game must be stated precisely.

In keeping with standard nomenclature in stochastic dynamic game theory we identify the two players as player I, the tracker, and player II, the evader.

Information known to one or both players:

i) The time interval of play, a fixed interval $[0,T]$, is known by both players.

ii) The dynamical description of the target (the evader's vehicle), an n dimensional linear system with state equation,

$$\dot{x} = F(t)x + G(t)u \tag{1}$$

where F and G denote continuous time function with appropriate dimensions and u is the control vector. The system structure and the matrices F and G are known by both players.

iii) The initial state of the target $x(0)$ is a zero mean Gaussian random vector with positive definite covariance matrix S. The statistics of the initial state are known by both players.

iv) Player I makes the following observations:

$$Z(t) = H(t)x + v(t), \tag{2}$$

$$t \in [0,T]$$

where: $H(t)$, a continuous time function with appropriated dimensions, is known by both players.

$v(t)$ is a zero mean Gaussian noise with covariance function

$$R(t)\,\delta(t-\tau),$$

$R(t) > \underline{0}$, $\forall t \in [0,T]$. The statistical description of $v(t)$ is known to both players. The symbol $\underline{Z}$ will denote the data set $\{Z(t) : 0 \leq t \leq T\}$.

v) The random variables $\{x(0)$, and $v(t)\}$ are statistically independent (this requirement reflects the physical independence of the sensor noise from the initial state of the target).

vi) The control function u is unknown to the tracker. The following information is common to both players:

The evader's choice of u is functionally independent of the random variables $\{x(0), v(t);\ t \in [0,T]\}$. The components of u are elements of $L^2[0,T]$. We denote this space of admissible controls by Ω. The following convention will be used in the sequel: U will denote the control function in order to avoid possible confusion between the function u and its value u(t) at time t.

We seek minimax estimates $\Psi(\underline{Z})$ for x(T) with respect to two exponentiated generalized quadratic loss functions defined by

$$L^+(\Psi,U) = e^{\frac{1}{2}(\Psi(\underline{Z})-x(T))'C(\Psi(\underline{Z})-x(T)) - \frac{1}{2}\int_0^T u'(t)D(t)u(t)dt} \tag{3}$$

$$L^-(\Psi,U) = -e^{-\frac{1}{2}(\Psi(\underline{Z})-x(T))'C(\Psi(\underline{Z})-x(T)) + \frac{1}{2}\int_0^T u'(t)D(t)u(t)dt} \tag{4}$$

based on the observation data $\underline{Z}$, $\underline{Z} \overset{\Delta}{=} \{Z(t) : t \in [0,T]\}$. C and D(t) are positive definite weighting matrices with appropriate dimensions. The characteristics of $L^+(\Psi,U)$ and $L^-(\Psi,U)$:

A continuous time minimax terminal state estimation problem for a linear plant under a generalized quadratic loss function has been considered previously by Mintz [3] and extended by Hexner and Mintz [4]; in both cases the following loss function was used,

$$L_o(\Psi,U) = \tfrac{1}{2}(\Psi(\underline{Z}) - x(T))C(\Psi(\underline{Z}) - x(T)) - \tfrac{1}{2}\int_0^T u'(t)D(t)u(t)dt. \tag{5}$$

This loss function forces the evader to compromise between a gain due to the estimation error, $\frac{1}{2}(\Psi(\underline{Z}) - x(T))'C(\Psi(\underline{Z}) - x(T))$, and a loss equal to $-\frac{1}{2}\int_0^T u'(t)D(t)u(t)dt$ due to the choice of the control U.

The loss function L^+, L^- and L_o are related as follows:

$$L^+(\Psi,U) = e^{L_o(\Psi,U)} \tag{6}$$

$$L^-(\Psi,U) = -e^{-L_o(\Psi,U)} \tag{7}$$

The loss functions L^+ and L^- exhibit a product structure, namely:

$$L(\Psi,U) = f(\Psi(\underline{Z}) - x(T))g(U) \tag{8}$$

in contrast to the difference structure exhibited by L_o.

We compare the behavior of L^+, L^- and L_o as follows:

a) For a fixed control U:

$$L^{+}(\Psi,U) \propto e^{\frac{1}{2}(\Psi(\underline{Z}) - x(T))'C(\Psi(\underline{Z}) - x(T))} \qquad (9a)$$

$$L^{-}(\Psi,U) \propto -e^{-\frac{1}{2}(\Psi(\underline{Z}) - x(T))C(\Psi(\underline{Z}) - x(T))} \qquad (9b)$$

and

$$L_o(\Psi,U) - \underset{\text{function}}{\text{constant}} \propto \tfrac{1}{2}(\Psi(\underline{Z}) - x(T)'C(\Psi(\underline{Z}) - x(T)). \qquad (9c)$$

b) For a fixed estimation error:

$$L^{+}(\Psi,U) \propto e^{-\frac{1}{2}\int_{o}^{T} u'(t)D(t)u(t)dt} \qquad (10a)$$

$$L^{-}(\Psi,U) \propto -e^{\frac{1}{2}\int_{o}^{T} u'(t)D(t)u(t)dt} \qquad (10b)$$

and

$$L_o(\Psi,U) - \underset{\text{function}}{\text{constant}} \propto -\int_{o}^{T} u'(t)D(t)u(t)dt. \qquad (10c)$$

In comparing $L^{+}(\Psi,U)$ with $L_o(\Psi,U)$ we see that $L^{+}(\Psi,U)$ accentuates the tracker's loss $[L_o(\Psi,U) > 0]$ and discounts the evader's loss $[L_o(\Psi,U) < 0]$ whereas $L^{-}(\Psi,U)$ discounts the tracker's loss $[L_o(\Psi,U) > 0]$ and accentuates the evader's loss $[L_o(\Psi,U) < 0]$.

We seek minimax terminal state estimates in each case in order to develop new insight into the existence, structure, and behavior of non-quadratic loss functions with a product structure, which provides an alternative to the use of generalized quadratic loss for modeling the evader compromise between his gain due to the estimation error and his loss due to the cost of control.

3. THEOREMS I AND II

We will use Lemmas 1 and 2 to establish Theorems I and II.

Lemma 1

Let the estimate $\Psi(\underline{Z})$ for $x(T)$ to be defined by the following Kalman filter equations: Define

$$\Psi(\underline{Z}) \overset{\Delta}{=} \hat{x}(T) = E\{x(T)|\underline{Z}\}$$

which by Sherman's theorem [1] is the Bayes estimate for $x(T)$, where

$$\dot{\hat{x}} = (F - KH)\hat{x} + KZ \qquad ; \ \hat{x}(0) = 0 \qquad (11a)$$

$$K = PH'R^{-1} \qquad (11b)$$

$$\dot{P} = FP + PF' - PH'R^{-1}HP \qquad ; \ P(0) = S \qquad (11c)$$

and P(t) > 0 for all t, since S > 0. Let $\epsilon(t)$ be defined by

$$\epsilon(t) = \hat{x}(t) - x(t). \tag{12}$$

From (1) and (11a) we get:

$$\dot{\epsilon} = (F - KH)\epsilon - Gu + Kv \quad ; \ \epsilon(0) = -x(0) \tag{13}$$

$\epsilon(t)$ can also be expressed by

$$\epsilon(t) = \epsilon_1(t) + \epsilon_2(t) \tag{14}$$

where

$$\dot{\epsilon}_1 = (F - KH)\epsilon_1 - Gu \quad , \ \epsilon_1(0) = 0 \tag{15}$$

$$\dot{\epsilon}_2 = (F - KH)\epsilon_2 + Kv \quad , \ \epsilon_2(0) = -x(0). \tag{16}$$

From the definitions of $\epsilon_1(t)$ and $\epsilon_2(t)$ we conclude that $\epsilon(T)$ is a normal random vector with mean $\mu = \epsilon_1(T)$ and covariance P(T), i.e., $\epsilon(T) \sim N[\epsilon_1(T), P(T)]$, where the differential equation of P is equation (11c).

The loss functions $L^+(\Psi,U)$ and $L^-(\Psi,U)$ are expressed in terms of ϵ as follows:

$$L^+(\Psi,U) = e^{\frac{1}{2}\epsilon' C\epsilon - \frac{1}{2}\int_0^T u'(t)D(t)u(t)dt} \tag{17}$$

$$L^-(\Psi,U) = -e^{-\frac{1}{2}\epsilon' C\epsilon + \frac{1}{2}\int_0^T u'(t)D(t)u(t)dt} \tag{18}$$

Lemma 2

Let Λ denote the class of all abstract probability measures that can be defined on Ω. Let $\lambda \in \Lambda$ be defined on the measurable space (Ω,Σ), where Σ denotes some σ-field of subsets of Ω. Assume that max $r[\hat{x}(T),U]$ exists, and denote its value by $\bar{r}$, where $\hat{x}(T)$ is the Bayes estimate for x(T) with respect to λ. Define $\Gamma \subset \Omega$ by

$$\Gamma = \{U : r[\hat{x}(T),U] = \bar{r}\} \tag{19}$$

where

$$r[\hat{x}(T),U] = E\{L(\Psi,U) \mid U\}.$$

If $\Gamma \in \Sigma$ and $\lambda[\Gamma] = 1$, then $\hat{x}(T)$ is a minimax estimate for x(T). This result is a statement in the context of our problem of a decision theoretic result due to Wald.

Theorem I

If i) $(C - P^{-1}) < 0$

and ii) the solution to the matrix Riccati equation,

$$\dot{R}_1 = -(F-KH)'R_1 - R_1(F-KH) - R_1GD^{-1}G'R_1 \tag{20}$$

$$R_1(T) = C + C(P^{-1}-C)^{-1}C \tag{21}$$

exists for all t in $[0,T]$, then $\hat{x}(T)$ defined by equation (11a) is a minimax estimate for $x(T)$ under the loss function $L^{+}(\Psi,U)$ and the risk function associated with $\hat{x}(T)$ is given by

$$r^{+}(\hat{x}(T),U) = e^{-\frac{1}{2}[\int_0^T u'Dudt - \varepsilon_1'(C+C(P^{-1}-C)^{-1}C)\varepsilon_1]} \left(\frac{\det(P^{-1}-C)^{-1}}{\det P}\right)^{\frac{1}{2}}. \tag{22}$$

Theorem I is proven in Section 4.

Theorem II

If the solution to the matrix Riccati equation

$$\dot{R}_2 = -(F-KH)'R_2 - R_2(F-KH) - R_2GD^{-1}GR_2 \tag{23}$$

$$R_2(T) = C - C(C+P^{-1})^{-1}C \tag{24}$$

exists for all t in $[0,T]$, then $\hat{x}(T)$ defined by equation (11a) is a minimax estimate for $x(T)$ under the loss function $L^{-}(\Psi,U)$ and the risk function associated with $\hat{x}(T)$ is given by,

$$r^{-}(\hat{x}(T),U) = -e^{\frac{1}{2}[\int_0^T u'Dudt - \varepsilon_1'(C-C(C+P^{-1})^{-1}C)\varepsilon_1]} \left(\frac{\det(P^{-1}+C)^{-1}}{\det P}\right)^{\frac{1}{2}}. \tag{25}$$

The proof of Theorem II is similar to that of Theorem I and is therefore omitted.

4. PROOF OF THEOREM I

Calculation of the risk function with $L^{+}(\Psi,U)$:

a) If $U = \theta$, i.e. $\{u(t) = 0, \ \forall t \in [0,T]\}$ then

$$L^{+}(\Psi,U)\Big|_{U=\theta} = e^{\frac{1}{2}\varepsilon'C\varepsilon} \tag{26}$$

and

$$r^{+}[\hat{x}(T),U]\Big|_{U=\theta} = \int_{-\infty}^{\infty} \frac{e^{\varepsilon'(\frac{C-P^{-1}}{2})\varepsilon} \, dt}{(2\pi)^{n/2}(\det P)^{\frac{1}{2}}} \tag{27}$$

where P denotes P(T) in (27).

For the risk function of equation (27) evaluated at $U=\theta$, two cases must be considered:

i) If $(\frac{C-P^{-1}}{2})$ has any nonnegative eigenvalues, the value of the risk function is infinite, and it can be shown that any decision rule is minimax.

ii) If $(\frac{C-P^{-1}}{2}) < 0$, the value of the risk function at $U=\theta$ is,

$$r^{+}[\hat{x}(T),U]\Big|_{U=\theta} = \left(\frac{\det(P^{-1}-C)^{-1}}{\det P}\right)^{\frac{1}{2}}. \tag{28}$$

b) If $U\neq\theta$ then by representing $\epsilon(t)$ by equation (14), the risk function is:

$$r^{+}[\hat{x}(T),U] = \frac{e^{-\frac{1}{2}\int_{0}^{T} u'Dudt}}{(2\pi)^{n/2}(\det P)^{\frac{1}{2}}}\int_{-\infty}^{\infty} e^{\epsilon_1'\frac{C}{2}\epsilon_1+\epsilon_1'\frac{C}{2}\epsilon_2+\epsilon_2'\frac{C}{2}\epsilon_1+\epsilon_2'\frac{(C-P^{-1})}{2}\epsilon_2} d\epsilon_2. \tag{29}$$

Let $(P^{-1}-C)=W$, and consider only the exponent in the integral, we have:

$$\epsilon_2'W\epsilon_2-\epsilon_1'C\epsilon_1-\epsilon_1'C\epsilon_2-\epsilon_2'C\epsilon_1 = (W^{\frac{1}{2}}\epsilon_2-W^{-\frac{1}{2}}C\epsilon_1)'(W^{\frac{1}{2}}\epsilon_2-W^{-\frac{1}{2}}C\epsilon_1)-\epsilon_1'(C+CW^{-1}C)\epsilon_1. \tag{30}$$

Substituting (29) into (30) we have:

$$r^{+}(\hat{x}(T),U) = \frac{e^{-\frac{1}{2}[\int_{0}^{T} u'Du - \epsilon_1'(C+C(P^{-1}-C)^{-1}C)\epsilon_1]}}{(2\pi)^{n/2}(\det P)^{\frac{1}{2}}}\cdot I_1 \tag{31}$$

where

$$I_1 = \int_{-\infty}^{\infty} e^{-\frac{1}{2}(\epsilon_2-W^{-1}C\epsilon_1)'W(\epsilon_2-W^{-1}C\epsilon_1)} d\epsilon_2 = (2\pi)^{n/2}(\det W^{-1})^{\frac{1}{2}}. \tag{32}$$

Substituting (32) into (31), we obtain:

$$r^{+}[\hat{x}(T),U] = e^{-\frac{1}{2}[\int_{0}^{T} u'Dudt-\epsilon_1'(C+C(P^{-1}-C)^{-1}C)\epsilon_1]}\left(\frac{\det(P^{-1}-C)^{-1}}{\det P}\right)^{\frac{1}{2}}. \tag{33}$$

Now, in order to show that $\hat{x}(T)$ is a minimax estimate for $x(T)$ when the conditions of Theorem I are satisfied, we consider the following maximization problem:

$$\max_{U} J_1[U]$$

where

$$J_1[U] = \epsilon_1'[C+C(P^{-1}-C)^{-1}C]\epsilon_1 - \int_{0}^{T} u'Dudt \tag{34}$$

$$\dot{\epsilon}_1 = (F-KH)\epsilon_1 + Gu \quad ; \; \epsilon_1(0) = 0. \tag{35}$$

To obtain equation (33) we assume $(P^{-1}-C) > 0$ which implies $(P^{-1}-C)^{-1} > 0$. Now since C and $(P^{-1}-C)^{-1}$ are positive definite, it follows that $(C + C(P^{-1}-C)^{-1}C)$ is positive definite; therefore, it follows from a standard control theoretic result, namely the existence of a bounded solution to the matrix Riccati equation,

$$R_1 = -(F-KH)'R_1 - R_1(F-KH) - R_1GD^{-1}G'R \tag{36a}$$

$$R_1(T) = C + C(P^{-1}-C)^{-1}C \tag{36b}$$

on the interval $[0,T]$, that the maximum of $J_1[U]$ occurs at $U=\theta$ and equals zero. In summary we have:

$$\max_U r^+[\hat{x}(T),U] = \left(\frac{\det(P^{-1}-C)^{-1}}{\det P}\right)^{\frac{1}{2}} . \tag{37}$$

Now by defining λ to be the degenerate probability distribution which assigns probability one to $U=\theta$, and by identifying $\Gamma = \{U : r^+[\hat{x}(T),U] = \bar{r}\} = \{\theta\}$ we can conclude from Lemma 2 that $\hat{x}(T)$ is minimax as $\hat{x}(T)$ is Bayes with respect to λ.

5. CONCLUSIONS

Theorems I and II provide sufficient conditions for $\hat{x}(T)$ to be a minimax estimate for $x(T)$ under the respective loss functions $L^+(\Psi,U)$ and $L^-(\Psi,U)$. It was shown that if the respective Riccati equations for each case have a bounded solution in the interval $[0,T]$, the Kalman filter provides a minimax estimate for $x(T)$ and the risk function $r[\hat{x}(T),U]$ will attain its maximum value at $U=\theta$. Hence the tracker will choose $\hat{x}(T)$ to be his minimax estimate for $x(T)$ and the evader will choose $u(t)$ equal to zero on the interval $[0,T]$ to be his least favorable control strategy.

If the Riccati equation exhibits finite escape at an interior point of $[0,T]$, then the Kalman filter will not provide a minimax estimate for $x(T)$. In this situation the Kalman filter will have an unbounded conditional risk function.

In comparing the results obtained by Mintz [3] (a dynamic terminal state estimation problem with a generalized quadratic loss function), Stelte and Mintz [5] (a static scalar problem with exponentiated generalized quadratic loss functions), and the cases considered in this paper, namely two dynamic terminal state estimation problems with exponentiated generalized quadratic loss functions, we note that in the three decision problems, the risk function attains a maximum with respect to the underlying parameter space when the unknown parameter is the zero vector. We further note that in Mintz [3] and our present case, the Kalman filter provides a minimax estimate for x(T).

Further work is necessary to determine the existence, structure and behavior of minimax terminal state estimates for the present class of problems with exponentiated generalized quadratic loss function when the length of the time interval is arbitrarily long.

REFERENCES

1. H. L. Van Trees, Detection, Estimation and Modulation Theory, Wiley and Sons, New York, 1968.
2. R. E. Kalman and R. S. Bucy, "New Results in Linear Filtering and Prediction Theory," ASME Journal of Basic Engineering, Vol. 83, pp. 95-107, 1961.
3. M. Mintz, "A Kalman Filter as a Minimax Estimator," Journal of Optimization Theory and Applications, Vol. 9, No. 2, pp. 99-111, February, 1972.
4. G. Hexner and M. Mintz, "On the Solution of a Minimax Terminal State Estimation Problem - A Hilbert Space Approach," Proceedings of the 1973 IEEE Conference on Decision and Control, pp. 645-649.
5. D. J. Stelte and M. Mintz, "On Minimax Parameter Estimation Under Exponentiated Generalized Quadratic Loss," Proceedings of the Eleventh Annual Allerton Conference on Circuit and System Theory, Univ. of Ill., pp. 215-225, October, 1973.

AN EXTENSION OF RADNER'S THEOREM TO NONZERO-SUM GAMES

TAMER BASAR
Applied Mathematics Division
Marmara Research Institute
Gebze-Kocaeli, Turkey

ABSTRACT

Radner's by now well-known theorem states that under rather mild conditions of existence, a static linear-quadratic-gaussian (LQG) team problem admits a unique linear globally optimal solution. In this paper, we extend this result to 2-person LQG nonzero-sum games with static information structures and obtain sufficient conditions which will ensure existence of unique affine Nash equilibrium solutions. These optimal strategies are related to the unique solution of a Lyapunov-type matrix equation.

I. INTRODUCTION

The decision problem considered in this paper is a 2-person nonzero-sum stochastic game defined by the quadratic cost functions

$$J_1(x,u,v)=u^TC_1x+\frac{1}{2}u^TD_{11}u+u^TD_{12}v\ , \tag{1a}$$

$$J_2(x,u,v)=v^TC_2x+\frac{1}{2}v^TD_{22}v+v^TD_{21}u\ , \tag{1b}$$

where C_i, D_{ij} are constant matrices of appropriate dimensions with $D_{ii}>0$, and x is an n- dimensional gaussian random vector with mean $\bar{x}$ and covariance $Q\geq 0$. Players 1 and 2 are provided with observations z_1 and z_2, respectively, which are given by

$$z_1=H_1x+w_1 \tag{2a}$$

$$z_2=H_2x+w_2\ , \tag{2b}$$

where w_1 and w_2 are statistically independent gaussian vectors of dimensions m_1 and m_2, respectively, with means zero and covariances $R_1>0$ and $R_2>0$. These random vectors are also statistically independent of x. H_1 and H_2 are real-valued matrices of appropriate dimensions.

In the description given above, u and v represent control variables of players 1 and 2, respectively. Player 1 is allowed to pick $u=\gamma_1(z_1)$ where $\gamma_1(.)$ is any r_1-dimensional Borel-measurable function of z_1, which is also square-integrable componentwise with respect to the appropriate gaussian measure which we denote by P_{z_1}. Analogously, Player 2 is allowed to pick $v=\gamma_2(z_2)$ where $\gamma_2(.)$ is any r_2- dimensional Borel-measurable function of z_2, which is also square-integrable componentwise with respect to the appropriate gaussian measure P_{z_2}.

Denoting the class of all permissible control laws γ_i for Player i by $L_{r_i}^{z_i}$, we define the expected (average) cost for Player i, for any fixed $\gamma_1\varepsilon L_{r_1}^{z_1}$, $\gamma_2\varepsilon L_{r_2}^{z_2}$, by

$$\bar{J}_i(\gamma_1,\gamma_2)=E[J_i(x,u,v)\,|\,u=\gamma_1(.),\ v=\gamma_2(.)], \tag{3}$$

where E[.] denotes the expectation operation over the prior statistics of the random variables x, w_1 and w_2. It can be shown that $\bar{J}_i(\gamma_1,\gamma_2)$ as given by (3) is well-defined and finite for every pair $\{\gamma_1 \varepsilon L^{z_1}_{r_1},\ \gamma_2 \varepsilon L^{z_2}_{r_2}\}$ [see [2]].

Within this set-up, we investigate, in this paper, existence of a *unique* equilibrium solution $\{\gamma_1^* \varepsilon L^{z_1}_{r_1},\ \gamma_2^* \varepsilon L^{z_2}_{r_2}\}$ satisfying the inequalities

$$\bar{J}_1(\gamma_1^*,\gamma_2^*) \leq \bar{J}_1(\gamma_1,\gamma_2^*) \tag{4a}$$

$$\bar{J}_2(\gamma_1^*,\gamma_2^*) \leq \bar{J}_2(\gamma_1^*,\gamma_2) \tag{4b}$$

for all $\gamma_i \varepsilon L^{z_i}_{r_i}$. What these inequalities roughly imply is that each player seeks a permissible decision rule so as to minimize his own average cost against some optimal policy of the other player.If the functional forms of the two cost functions are equivalent, then we have a 2-person team problem in which each member has different information but strives for the same goal. One of the most important results in this area is Radner's theorem on teams with static information structures, which roughly states that under rather mild conditions of existence a static LQG team problem admits a unique globally optimal linear solution [8]. Not much has been done since then in extending this result to nonzero-sum stochastic games. Only recently Basar and Ho have shown existence of a unique Nash equilibrium point in linear strategies, for a special class of LQG problems in which the control variables are scalar [3]. In the sequel,we will take a different approach than the one taken in [3] and derive conditions under which the stochastic nonzero-sum game of this section will admit unique affine Nash equilibrium strategies.

II. EXISTENCE CONDITIONS FOR A UNIQUE EQUILIBRIUM SOLUTION

The space $L^{z_i}_{r_i}$, which has been defined earlier in section I, can be turned into a metric space by defining the norm

$$||\gamma||^{z_i}_{r_i} \triangleq \{\int_{R^{m_i}} \gamma^T(z_i)\gamma(z_i)\,dP_{z_i}(z_i)\}^{1/2} \quad , \tag{5}$$

where the subscript r_i stands for the dimension of the vector-valued function $\gamma \varepsilon L^{z_i}_{r_i}$ and the superscript z_i identifies the gaussian probability measure P_{z_i}. In fact (5) is only a semi-norm on $L^{z_i}_{r_i}$, but can be made into a norm by passing into equivalence classes with respect to P_{z_i}. Denoting the subset of $L^{z_i}_{r_i}$, which is the collection of these equivalence classes of functions, by $\mathcal{L}^{z_i}_{r_i}$, expression (5) now defines a norm on this new space and $\mathcal{L}^{z_i}_{r_i}$ becomes a linear space.It can further be shown that $\mathcal{L}^{z_i}_{r_i}$ is also *complete* i.e., it is a *Banach* space [see e.g. Ash [1] p.85].

For any fixed $\gamma_j \varepsilon \mathcal{L}^{z_j}_{r_j}$, the unique optimal element $\gamma_i^* \varepsilon \mathcal{L}^{z_i}_{r_i}$ is given by [see e.g.Kushner [5], p.245]

$$\gamma_i^*(z_i) = -D_{ii}^{-1}C_iE[x|z_i] - D_{ii}^{-1}D_{ij}E[\gamma_j(z_j)|z_i] \ ,\quad i \neq j\ ; i,j=1,2, \tag{6}$$

where $E[f(.)|z_i]$ denotes the expected value of $f(.)$conditioned on the observed value of the random vector z_i. Now, substitution of γ_j^* as given by (6) into the expression for γ_i^*, $i \neq j$, results in

$$\gamma_i^*(z_i) = -D_{ii}^{-1}C_iE[x|z_i] + D_{ii}^{-1}D_{ij}D_{jj}^{-1}C_jE[E[x|z_j]|z_i]$$

$$+D_{ii}^{-1}D_{ij}D_{jj}^{-1}D_{ji}E[E[\gamma_i^*(z_i)|z_j]|z_i] \tag{7a}$$

$$\triangleq K^i\ \gamma_i^*(z_i) = K_1^i x + K_2^i\ \gamma_i^*(z_i) \quad , \tag{7b}$$

where $K_1^i x$ represents the first two terms of (7a) and K_2^i represents the linear operator

$$D_{ii}^{-1}D_{ij}D_{jj}^{-1}D_{ji}\ E[E[\cdot|z_j]|z_i].$$

It is important to note that the closed-loop relations (7) represent the complete class of Nash equilibrium solutions to the problem. Hence, we now seek conditions under which there will exist a unique solution to (7). We first need the following lemma:

Lemma 1:

Let z_1, z_2 be random vectors defined as before. Then for any Borel-measurable function $\gamma_j \varepsilon \mathcal{L}_{r_j}^{z_j}$, the linear operator ε_i defined by

$$\varepsilon_i\gamma_j \triangleq E\ [\gamma_j(z_j)|z_i] \tag{8a}$$

is a *nonexpansive* transformation from $\mathcal{L}_{r_j}^{z_j}$ into $\mathcal{L}_{r_j}^{z_i}$, i.e. for any $\gamma,\ \xi\ \varepsilon\ \mathcal{L}_{r_j}^{z_j}$,

$$||\varepsilon_i\gamma - \varepsilon_i\xi||_{r_j}^{z_i} \leq ||\gamma-\xi||_{r_j}^{z_j} \quad . \tag{8b}$$

Proof of Lemma 1:

The proof will follow from the following sequence of inequalities, where γ_j^k will denote the k'th component of the vector-valued function γ_j:

$$(\varepsilon_i\gamma_j)^T(\varepsilon_i\gamma_j) = \sum_{k=1}^{r_j}\{E[\gamma_j^k(z_j)|z_i\ ]\}^2$$

$$\leq \sum_{k=1}^{r_j}E[\{\gamma_j^k(z_j)\}^2|z_i]$$

where the last expression follows from the well-known Cauchy-Schwartz inequality [see e.g.Ash [1], p. 82] .Therefore we have

$$\{||\varepsilon_i\gamma_j||_{r_j}^{z_i}\}^2 = \sum_{k=1}^{r_j}\int_{R^{m_i}}\{E[\gamma_j^k(z_j)|z_i]\}^2\ dP_{z_i}\ (z_i)$$

$$\leq \sum_{k=1}^{r_j}\int_{R^{m_i}}E[[\gamma_j^k(z_j)\ ]^2|z_i\ ]dP_{z_i}\ (z_i).$$

But since expectation is the expectation of a conditional expectation, the last expression is identical to

$$\sum_{k=1}^{r_j}\int_{R^{m_j}}[\gamma_j^k(z_j)]^2\ dP_{z_j}\ (z_j) = \{||\gamma_j||_{r_j}^{z_j}\ \}^2$$

Hence we have proven

$$||\varepsilon_i \gamma_j||_{r_j}^{z_i} \le ||\gamma_j||_{r_j}^{z_j} \quad . \tag{8c}$$

But since $\mathcal{L}_{r_j}^{z_i}$ is a linear space and ε_i is a linear operator, inequality (8c) is equivalent to inequality (8b).

Q.E.D.

The following theorem now establishes the desired result on uniqueness of the Nash solution:

Let $\bar{\lambda}(\pi)$ denote a maximum eigenvalue of the bounded symmetric matrix $\pi^T\pi$. Then, we have:

Theorem 1:

If either

$$\text{(i)} \quad \bar{\lambda}(D_{11}^{-1}D_{12}D_{22}^{-1}D_{21})<1 \tag{9a}$$

or

$$\text{(ii)} \quad \bar{\lambda}(D_{22}^{-1}D_{21}D_{11}^{-1}D_{12})<1 \quad , \tag{9b}$$

then there exists a unique pair of Borel-measurable $\gamma_i \varepsilon \mathcal{L}_{r_i}^{z_i}$ that solves equation (6).

Proof: The proof will be completed if we can show that either K^1 defined via (7b) is a contraction mapping from $\mathcal{L}_{r_1}^{z_1}$ into $\mathcal{L}_{r_1}^{z_1}$ or K^2 is a contraction mapping from $\mathcal{L}_{r_2}^{z_2}$ into $\mathcal{L}_{r_2}^{z_2}$ under either one of the conditions (9a) or (9b), since then the theorem follows from Banach's classical contraction mapping principle (fixed-point theorem) [see e.g.[6], p.27].

Taking i=1, it can first of all be shown that $K_1^1 x \varepsilon \mathcal{L}_{r_1}^{z_1}$ and that K_2^1 is a linear operator mapping $\mathcal{L}_{r_1}^{z_1}$ into $\mathcal{L}_{r_1}^{z_1}$. Since $\mathcal{L}_{r_1}^{z_1}$ is a linear space, it follows that K^1 maps $\mathcal{L}_{r_1}^{z_1}$ into $\mathcal{L}_{r_1}^{z_1}$. Defining a metric $d(\gamma,\xi)$ on this space by the norm $||\gamma-\xi||_{r_1}^{z_1}$,we show that K^1 is a contraction mapping as follows:

For every $\gamma, \xi \; \varepsilon \; \mathcal{L}_{r_1}^{z_1}$

$$d(K^1\gamma,K^1\xi)=||K^1\gamma-K^1\xi||_{r_1}^{z_1} =||K_2^1(\gamma-\xi)||_{r_1}^{z_1}$$

$$\le [\bar{\lambda}(D_{11}^{-1}D_{12}D_{22}^{-1}D_{21})]^{1/2}||E[E[(\gamma-\xi)|z_2]|z_1||_{r_1}^{z_1}$$

where we have used the fact that

$$(D_{11}^{-1}D_{12}D_{22}^{-1}D_{21})^T(D_{11}^{-1}D_{12}D_{22}^{-1}D_{21}) \le \bar{\lambda}(D_{11}^{-1}D_{12}D_{22}^{-1}D_{21})I_{r_1} \quad .$$

Using the notation of Lemma 1 we can write the last expression as

$$[\bar{\lambda}(D_{11}^{-1}D_{12}D_{22}^{-1}D_{21})]^{1/2}||\varepsilon_1\varepsilon_2(\gamma-\xi)||_{r_1}^{z_1} \quad .$$

Since ε_i is a nonexpansive transformation by Lemma 1,

$$||\varepsilon_1\varepsilon_2(\gamma-\xi)\;||_{r_1}^{z_1} \le ||\;\varepsilon_2(\gamma-\xi)||_{r_2}^{z_2} \le ||\gamma-\xi||_{r_1}^{z_1} = d(\gamma,\xi)$$

Hence, we have shown that

$$d(K^1\gamma, K^1\xi) \leq [\bar{\lambda}(D_{11}^{-1}D_{12}D_{22}^{-1}D_{21})]^{1/2}\ d(\gamma,\xi).$$

Using condition (9a), this last inequality implies that K^1 is a contraction mapping, which in turn establishes existence of a unique fixed point by direct application of Banach's contraction mapping principle, since $\mathcal{L}^{z_1}_{r_1}$ is a complete metric space.

Since relation (6) determines $\gamma_2^*(z_2)$ uniquely for every fixed $\gamma_1(.)\varepsilon\,\mathcal{L}^{z_1}_{r_1}$, existence and uniqueness of γ_1^* also implies existence and uniqueness of $\gamma_2^*\varepsilon\,\mathcal{L}^{z_2}_{r_2}$. Hence we have proven that under condition (9a) there exists a unique Nash equilibrium solution.

A similar contraction result can be obtained for K^2 under condition (9b), and this completes the proof of Theorem 1.

Q.E.D.

III. EXACT EXPRESSIONS FOR THE UNIQUE NASH STRATEGIES

We now show that the unique Nash strategies given by Theorem 1 and the closed-loop relations (7) can actually be explicitly computed and shown to be affine in the observation of each player. These results are summarized in Theorem 2 below, a proof of which is provided in the sequel.

Theorem 2:

For the nonzero-sum stochastic game introduced in section I and under either of the conditions (9a) or (9b) of Theorem 1, there exist *unique affine* Nash equilibrium strategies given by

$$u^*=\gamma_1^*(z_1)= A_1\bar{x}+B_1(z_1-H_1\bar{x}) \tag{10a}$$

$$v^*=\gamma_2^*(z_2)= A_2\bar{x}+B_2(z_2-H_2\bar{x}) \tag{10b}$$

where A_1 and A_2 are given by

$$A_i=-[I-D_{ii}^{-1}D_{ij}D_{jj}^{-1}D_{ji}]^{-1}D_{ii}^{-1}[C_i-D_{ij}D_{jj}^{-1}C_j] \quad i,j=1,2;i\neq j. \tag{11}$$

and B_1 is given as the unique solution of the Lyapunov-type equation

$$B_1+PB_1G = F \tag{12a}$$

with

$$P \triangleq -D_{11}^{-1}D_{12}D_{22}^{-1}D_{21} \tag{12b}$$

$$G \triangleq H_1QH_2^T(H_2QH_2^T+R_2)^{-1}H_2QH_1^T(H_1QH_1^T+R_1)^{-1} \tag{12c}$$

$$F \triangleq -D_{11}^{-1}C_1QH_1^T(H_1QH_1^T+R_1)^{-1}$$
$$+D_{11}^{-1}D_{12}D_{22}^{-1}C_2QH_2^T(H_2QH_2^T+R_2)^{-1}H_2QH_1^T(H_1QH_1^T+R_1)^{-1} \tag{12d}$$

and B_2 is given by

$$B_2 \triangleq -D_{22}^{-1}\ [C_2+D_{21}B_1H_1]QH_2^T(H_2QH_2^T+R_2)^{-1}\ . \tag{12e}$$

Proof of Theorem 2 :

Using the well-known result

$$E[z_i - H_i \bar{x} | z_j] = H_i Q H_j^T (H_j Q H_j^T + R_j)^{-1} (z_j - H_j \bar{x}), \quad i \neq j$$

in (7a), it is not difficult to see that $\gamma_1^*(z_1) = A_1 \bar{x} + B_1(z_1 - H_1 \bar{x})$ satisfies equation (7a) for any B_1 satisfying (12a). And further substitution of this solution into the expression for $\gamma_2^*(z_2)$ given by (6), provides us with the affine policy $A_2 \bar{x} + B_2(z_2 - H_2 \bar{x})$ where A_2 is given by (11) and B_2 by (12e).

In order to complete the proof of Theorem 2 we will still have to show that

(i) A_1 and A_2 given by (11) are well-defined under the condition of Theorem 1,

(ii) There exists at least one solution of (12a) under the same condition.

To establish the validity of the former, we first assume that condition (9a) holds true and note that this implies, via Lemma A-1 given in the Appendix, that $[I - D_{11}^{-1} D_{12} D_{22}^{-1} D_{21}]$ is nonsingular. If, however, condition (9a) is not satisfied but condition (9b) holds, then this implies via the same lemma that

$$|\lambda_{max}(D_{22}^{-1} D_{21} D_{11}^{-1} D_{12})| < 1 .$$

But since the nonzero eigenvalues of $D_{22}^{-1} D_{21} D_{11}^{-1} D_{12}$ and $D_{11}^{-1} D_{12} D_{22}^{-1} D_{21}$ are the same [[7], p.24] ,

$$|\lambda_{max}(D_{22}^{-1} D_{21} D_{11}^{-1} D_{12})| < 1 \text{ is equivalent to } |\lambda_{max}(D_{11}^{-1} D_{12} D_{22}^{-1} D_{21})| < 1.$$

Hence A_1 is well-defined under either of conditions (9a) or (9b). It follows from a similar reasoning that A_2 is also well-defined.

In proving the latter, we will in fact show directly that there exists a unique $(r_1 x m_1)$ matrix B_1 that satisfies (12a). For this purpose, define an $(r_1 m_1)$-dimensional vector $\underline{b}$ such that the ij'th entry of A_1 is the $[(i-1)m_1 + j]$'th element of $\underline{b}$ Define a vector $\underline{f}$ which is related to F in a similar manner. Then, equation (12a) can also be written as

$$(I + P \otimes G^T)\underline{b} = \underline{f} \tag{13}$$

where I is the $(r_1 m_1 x r_1 m_1)$ identity matrix, and $P \otimes G^T$ denotes the $(r_1 m_1 x r_1 m_1)$-dimensional Kronecker product of P and G^T. Equation (13) has a unique solution for all $\underline{f}$ if, and only if, $(I + P \otimes G^T)$ is nonsingular. Hence it will be sufficient to show that $|Re\lambda(P \otimes G^T)| < 1$ for all $\lambda(.)$. We know from a property of Kronecker products that [Bellman [4], p.235];

$$\lambda[P \otimes G^T] = \mu_j \beta_k$$

where μ_j is any eigenvalue of P and β_k is any eigenvalue of G^T. The proof will then be completed if we can show that $|\lambda_{max}(P)| < 1$ and $|\lambda_{max}(G^T)| < 1$. The former follows from the discussion given on the previous page in the proof of (i), and the latter follows from Lemma A-2 given in the Appendix.

IV. CONCLUSION

In this paper we have considered a class of LQG 2-person nonzero-sum stochastic games with static information structure. By making use of an appropriately structured Banach space, we have derived equilibrium solutions. It is important to note that in establishing this result we have not made use of the specific probabilistic structure of the problem. As it has been mentioned in a related paper [2], this uniqueness result is valid for a much broader class of decision problems.

We have also shown in the paper that for the LQG decision problem with static information structure, the above-mentioned unique equilibrium solution is realized by affine strategies for both players, which is an extension of Radner's theorem on LQG team problems to nonzero-sum games. For the game problem, these unique affine policies are related to the unique solution of a Lyapunov-type matrix equation.

APPENDIX

Lemma A-1 :

If A is a real-valued nxn square matrix, then $\lambda_{max}(A^TA)<1$ implies $|\lambda(A)|<1$ for all $\lambda(A)$, and hence $(I-A)$ nonsingular. [Here $\lambda(.)$ denotes any eigenvalue of the matrix (.) and $\lambda_{max}(.)$ denotes the maximum eigenvalue of the symmetric matrix (.).]

Proof: Note that for any n-dimensional complex-valued x,

$$\frac{\bar{x}^TA^TAx}{\bar{x}^Tx} \le \lambda_{max}(A^TA)<1 \ . \tag{A-1}$$

Now assume the contrary to the lemma, i.e. there exists an eigenvalue μ of A with $|\mu|\ge 1$. Let y be an eigenvector corresponding to μ, i.e. $Ay=\mu y$. Hence

$$\bar{y}^TA^TAy = (\overline{Ay})^T(Ay)=|\mu|^2\bar{y}^Ty \quad ,$$

which forms a contradiction to (A-1) if $|\mu|\ge 1$. This completes the proof of the lemma since the above implies $\mathrm{Re}\,\lambda(A)<1$ for all $\lambda(A)$.

Q.E.D.

Lemma A-2 :

When $Q\ge 0$, $R_1>0$, $R_2>0$ and H_1, H_2 are arbitrary real matrices of appropriate dimensions (H_1 and H_2 are not necessarily square), all eigenvalues of the square matrix G defined by (12c) are real, nonnegative and less than unity.

Proof : Let λ denote an eigenvalue of G^T and let x denote a corresponding eigenvector. Then

$$(H_1QH_1^T+R_1)^{-1}H_1QH_2^T(H_2QH_2^T+R_2)^{-1}H_2QH_1^Tx=\lambda x$$

which can also be written as

$$H_1QH_2^T(H_2QH_2^T+R_2)^{-1}H_2QH_1^Tx=(H_1QH_1^T+R_1)\lambda x. \tag{A-2}$$

Denoting $H_1QH_1^T$ by Q_1 for notational simplicity, and multiplying each side of (A-2) by $\bar{x}^T$,we have

$$\bar{x}^T H_1 Q H_2^T (H_2 Q H_2^T + R_2)^{-1} H_2 Q H_1^T x = \lambda \bar{x}^T (Q_1 + R_1) x \; . \qquad \text{(A-3)}$$

Now noting the relation $C^T(CC^T+R_2)^{-1}C<I$ for an arbitrary m_2xn matrix C, we identify C by $H_2Q^{1/2}$, and observe that the following inequality follows from (A-3):

$$\lambda\bar{x}^T(Q_1+R_1)x<\bar{x}^TQ_1x$$

which in turn is

$$\lambda\bar{x}^TR_1x<(1-\lambda)\bar{x}^TQ_1x \; .$$

Since $R_1>0$, $Q_1\geq 0$, this last inequality implies that $\mathrm{Im}\lambda =0$, $0\leq \mathrm{Re}\lambda<1$ for any $\lambda(G)$.

Q.E.D.

REFERENCES

[1] R.B.Ash, *Real Analysis and Probability*, New York;Academic Press, 1972.

[2] T.Basar, "Equilibrium Solutions in Multi-criteria Decision Problems with Static Information Structures" , submitted for publication.

[3] T.Basar and Y.C.Ho, "Informational Properties of the Nash Solutions of Two Stochastic Nonzero-Sum Games" , *Journal of Economic Theory*, Vol.7, No.4, pp.370-387, April 1974.

[4] R. Bellman, *Introduction to Matrix Analysis*, 2nd edition, New York; McGraw-Hill, 1970.

[5] H. Kushner, *Introduction to Stochastic Control*, New York; Holt, Rinehart and Winston, 1971.

[6] L.A. Liusternik and V.J. Sobolev, *Elements of Functional Analysis*, New York;Ungar, 1961.

[7] M. Marcus and H.Minc, *A Survey of Matrix Theory and Matrix Inequalities*, Vol.14, Boston;Prindle, Weber and Schmidt, 1964.

[8] R.Radner, " Team-Decision Problems" , *Ann.Math.Statist.*, Vol.33, No.3, pp.857-881, 1962.

A RECURSIVE ALGORITHM FOR IMAGE ENHANCEMENT

RODOLFO FEICK and MICHAEL G. STRINTZIS
Department of Electrical Engineering
University of Pittsburgh, Pittsburgh, Pennsylvania

ABSTRACT

A simple suboptimal linear algorithm for the enhancement of two-dimensional images corrupted by additive white noise is presented. The algorithm is of general applicability and offers substantial savings in computational and storage requirements over the optimal algorithms at the cost of a modest increase in the mean square error. As such it is well suited for on-line enhancement of images characterized by arbitrary but known two-dimensional covariance functions.

INTRODUCTION

In recent years a great deal of work [1], [2], [3] has been devoted to the task of finding recursive linear filters to enhance two-dimensional images corrupted by additive noise. Several successful algorithms have been described in the literature. All share the drawbacks of (a) requiring a considerable storage capacity as well as computational effort and (b) being only applicable to images characterized by a somewhat restricted class of covariance functions. In application to images with other types of covariance functions these methods require rather formidable storage and computational capability [4].

In this paper we explore the development of a computationally simple recursive linear filter for two-dimensional images corrupted by additive white noise. The filter consists of a linear estimator designed to obtain the best estimate for a point in terms of the measurement at the point and the estimates of a fixed, finite number of neighboring points which have already been processed. The criterion for optimality is minimization of the mean square error.

DESCRIPTION OF THE FILTER

Let $\{x(k,s)\}_{k=-\infty,s=-\infty}^{k=\infty,s=\infty}$ be a two dimensional, zero mean, wide sense stationary stochastic process with covariance function $R(i,j) = E[x(k,s)x(k+i,s+j)]$. We assume additive white observation noise $v(k,s)$, uncorrelated with the process.

$$y(k,s) = x(k,s) + v(k,s) \tag{1}$$

$$E[v(k,s)v(k+i,s+j)] = \begin{cases} V & \text{if } i = j = 0 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$$E[x(k,s)v(m,n)] = 0 \tag{3}$$

The form of the proposed filter is

$$\hat{x}(k,s) = a_1\hat{x}(k,s-1) + a_2\hat{x}(k-1,s) + a_3\hat{x}(k-1,s-1) + by(k,s) \tag{4}$$

where $\hat{x}(k,s)$ is the estimate of $x(k,s)$ (See Figure 1(a)). The choice of $\hat{x}(k-1,s)$, $\hat{x}(k,s-1)$, $\hat{x}(k-1,s-1)$ as the "past" estimates is arbitrary and is based only upon the intuitive appeal of the scheme. An alternate scheme, illustrated in Figure 1(b), is currently being investigated.

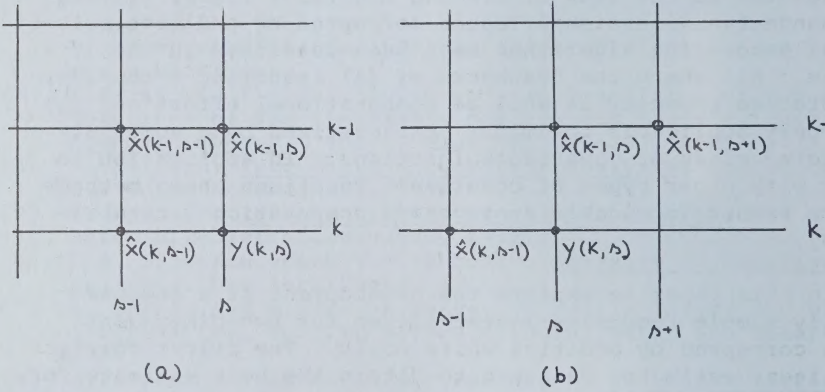


FIGURE 1 Examples of Estimation Schemes

For simplicity, we shall assume here that the "steady state" of the algorithm has been reached and, therefore, that the coefficients a_1, a_2, a_3 and b in (4) do not depend on k and s. Implementation of the scheme requires the determination of the coefficients which minimize the mean square error given data $\hat{x}(k-1,s)$, $\hat{x}(k,s-1)$, $\hat{x}(k-1,s-1)$ and $y(k,s)$. By virtue of the orthogonality principle a_1, a_2, a_3 and b satisfy:

$$E[(\hat{x}(k,s) - x(k,s))\hat{x}(k,s-1)] = 0 \tag{5}$$

$$E[(\hat{x}(k,s) - x(k,s))\hat{x}(k-1,s)] = 0 \qquad (6)$$

$$E[(\hat{x}(k,s) - x(k,s))\hat{x}(k-1,s-1)] = 0 \qquad (7)$$

$$E[(\hat{x}(k,s) - x(k,s))y(k,s)] = 0 \qquad (8)$$

On replacing $\hat{x}(k,s)$ by its value as given by (4) we obtain the following equations:

$$\begin{aligned} &a_1E[\hat{x}(k,s-1)\hat{x}(k,s-1)] + a_2E[\hat{x}(k-1,s)\hat{x}(k,s-1)] \\ &+ a_3E[\hat{x}(k-1,s-1)\hat{x}(k,s-1)] + bE[y(k,s)\hat{x}(k,s-1)] = \\ &\qquad E[x(k,s)\hat{x}(k,s-1)] \end{aligned} \qquad (9)$$

$$\begin{aligned} &a_1E[\hat{x}(k,s-1)\hat{x}(k-1,s)] + a_2E[\hat{x}(k-1,s)\hat{x}(k-1,s)] \\ &+ a_3E[\hat{x}(k-1,s-1)\hat{x}(k-1,s)] + bE[y(k,s)\hat{x}(k-1,s)] = \\ &\qquad E[x(k,s)\hat{x}(k-1,s)] \end{aligned} \qquad (10)$$

$$\begin{aligned} &a_1E[\hat{x}(k,s-1)\hat{x}(k-1,s-1)] + a_2E[\hat{x}(k-1,s)\hat{x}(k-1,s-1)] \\ &+ a_3E[\hat{x}(k-1,s-1)\hat{x}(k-1,s-1)] + bE[y(k,s)\hat{x}(k-1,s-1)] = \\ &\qquad E[x(k,s)\hat{x}(k-1,s-1)] \end{aligned} \qquad (11)$$

$$\begin{aligned} &a_1E[\hat{x}(k,s-1)y(k,s)] + a_2E[\hat{x}(k-1,s)y(k,s)] \\ &+ a_3E[\hat{x}(k-1,s-1)y(k,s)] + b(R(0,0) + V) = R(0,0) \end{aligned} \qquad (12)$$

The minimum mean square error can easily be shown to be equal to bV. The above is a system of highly nonlinear equations in a_1, a_2, a_3 and b with no known closed form solution. A computer program to iteratively solve these equations has been developed. The only information required to obtain the optimum coefficients is the covariance function of the image. No assumptions on the form of this function have been made.

RESULTS

The optimum coefficients for several representative cases were calculated. Two types of covariance functions were used

(a) $R(i,j) = R(0,0) \exp(-\sqrt{ci^2 + dj^2})$

(b) $R(i,j) = R(0,0) \exp(-ci - dj)$

Since the noise power after filtering is bV, the noise reduction ratio is 1/b and the performance of the filter in decibels is defined as

$$N = 10 \log_{10} (1/b) \text{ [db]}$$

The results are summarized in the following table:

R(0,0)/V	c	d	N	R(0,0)/V	c	d	N
0.83	0.25	0.25	4.4	0.83	0.5	0.5	4.4
0.83	0.04	0.04	5.8	0.83	0.2	0.2	5.9
0.83	0.00766	0.00766	7.4	0.83	0.0875	0.0875	7.7
$R(i,j) = R(0,0)\exp(-\sqrt{ci^2+dj^2})$				$R(i,j) = R(0,0)\exp(-ci-dj)$			

As can be seen a rather significant reduction of the noise can be achieved with this very simple on-line enhancement algorithm. The performance of the filter is only slightly inferior to that of the far more complicated optimal algorithms. In fact, in tests with noisy pictures having a covariance of type (b) with c = d = 0.0875, and a signal to noise power ratio as indicated in the above table, the performance of a vector scanning optimal filter has been reported [3] to be close to 8 [db]. It is to be noted that the best to-date reported filter performance corresponds to vector scanning algorithms; more conventional scalar scanning results in a slight degradation of the performance.

REFERENCES

[1] N.E. Nahi and T. Assefi, "Bayesian Recursive Image Estimation," IEEE Trans. Comput. (Short Note), Vol. C-21, pp. 734-738, July 1972.

[2] N.E. Nahi, "Role of Recursive Estimation in Statistical Image Enhancement," Proc. IEEE, Vol. 60, pp. 872-877, July 1972.

[3] N.E. Nahi and C. A. Franco, "Recursive Image Enhancement - Vector Processing," IEEE Trans. Comm., Vol. COM-19, pp. 305-311, April 1973.

[4] S.R. Powell and L. M. Silverman, "Modeling of Two-Dimensional Covariance Functions with Application to Image Restoration," IEEE Trans. Aut. Cont., Vol. AC-19, pp. 8-12, February 1974.

ON AN INTEGRAL EQUATION IN PHASE ESTIMATION

N. C. MOHANTY
Electrical Engineering Department
State University of New York at Buffalo
Buffalo, New York

1. Introduction

In the problems of phase estimation and signal detection, (Vantrees, 1968), we would like to solve

$$\sqrt{E}\, f(y) = \int_o^a u(y)\, k_n\, (x,y,)\, dy \quad 0 \lesssim x \lesssim a \tag{1.1}$$

where signal f(x) and k_n (x,y), the covariance of noise are known and u(.) is unknown. The noise contains a non-zero white noise term.

$$k_n\, (x_1 y) = \frac{N_o}{2}\, \delta\, (x-y) + k_c(x,y)$$

where k_c (x,y) is a continuous square integrable function. The equation (1.1) becomes

$$\sqrt{E}\, f(x) = \frac{N_o}{2}\, u(x) + \int_o^a k_c\, (x,y)\, u(y)\, dy, \quad o \lesssim x \lesssim a \tag{1.2}$$

The noise $n_c(t)$ is the steady state response of a lumped, linear passive network excited with white Gaussian noise. Here the covariance function is function ot its arguments, i.e.

$$k_c(x,y) = k_c(x-y)$$

Without loss of generality, we assume noise components are zero mean. We can further write

$$k_c(s,y) = k_c(x-y) = k(1x-y1)$$

and noise is further assumed to real random process. With these assumptions, the integral equation reduced to

$$\sqrt{E}\, f(x) = \frac{N_o}{2}\, u(x) + \int_o^a k(1x-y1)\, u(y)\, dy \quad o \lesssim x \lesssim a \tag{1.3}$$

After normalization (1.3) can be put to

$$u(x) = f(x) + \int_o^a k(1x-y1)\, dy, \quad o \lesssim x \lesssim a$$

We apply differential approximation method (Mohanty, 1971) to approximate the autocorrelation function

$$k(t) = \sum_{i=1}^{N} a_i e^{-b_i t}, \quad b_i \gtrsim o \text{ for all } i$$

This approximation ensures rational spectra and a stable solution. This is discussed in section 2. The integral equation is converted to a vector differential equation with two point boundary value problem (Bellman, 1967). In section 3, inverse Laplace transform method (Ogata, 1967) is applied to convert it to a initial value problem. Finally, we have given an example to illustrate our method.

2. Differential Approximation Method

We replace the kernel $K(|..|)$ by an approximating sum of the form

$$K(t) = \sum_{i=1}^{n} a_i e^{-b_i t} \tag{2.1a}$$

which will satisfy the n^{th} order differential equation

$$K^n(t) + c_1 K^{n-1}(t) + c_2 K^{n-2}(t) + \ldots + c_n = 0, \tag{2.1b}$$

where the coefficients c_i are to be determined from the equations which minimize the integral

$$\min_{c_i} \int_0^a (k^n(t) + c_1 K^{n-1}(t) + c_2 K^{n-2}(t) + \ldots + c_n)^2 dt. \tag{2.1c}$$

One finds the roots of $\lambda = b_i$ of the algebraic equation from

$$\sum_{i=0}^{n} c_i \lambda^{n-i} = 0, \; c_0 = 1 \tag{2.1d}$$

and a_i's are determined from the equation

$$\min_{a_i} \int_0^a (K(t) - \Sigma a_i e^{-b_i t})^2 dt \tag{2.1e}$$

By using (2.1a), equation (1.1) becomes

$$u(x) = f(x) + \sum_{i=1}^{n} a_i \left[\int_0^x e^{-b_i(x-y)} u(y)dy + \int_x^a e^{-b_i(y-x)} u(y)dy\right]$$

$$= f(x) + \mathop{}_{i=1}^{n} a_i(V_i(x) + W_i(x)), \tag{2.2}$$

where

$$\begin{aligned} v_i(x) &= \int_0^x e^{-b_i(x-y)} u(y)dy \\ w_i(x) &= \int_x^a e^{-b_i(y-x)} u(y)dy \;. \end{aligned} \tag{2.3}$$

Differentiating (2.3) with respect to x, we find

$$\begin{aligned} v_i'(x) + b_i v_i(x) &= u(x), \; v_i(0) = 0 \\ w_i'(x) - b_i w_i(x) &= -u(x), \; w_i(a) = 0 \end{aligned} \tag{2.4}$$

From the equns. (2.2) and 2.4), we get

$$\begin{aligned} v_i'(x) &= -b_i v(x) + f(x) + \sum_{i=1}^{n} a_i(v_i(x) + w_i(x)) \\ v_i(0) &= 0, \\ w_i'(x) &= b_i w_i(x) - f(x) - \sum_{i=1}^{n} a_i(v_i(x) + w_i(x)), \\ w_i(a) &= 0 \;. \end{aligned} \tag{2.5}$$

Hence, in the vector matrix form, these equations (2.5) can be written as

$$\begin{aligned} v'(x) &= f(x)\eta + By(x) + Cw(x), \quad v(0) = 0 \\ w'(x) &= f(x)\eta - Cv(x) - Bw(x), \quad 2(a) = 0 \end{aligned} \tag{2.6}$$

where

$$v = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \qquad w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} \qquad \eta = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

$$B = \begin{bmatrix} a_1-b_1 & 0 & 0 & 0 & 0 \\ a_1 & a_2-b_2 & 0 & 0 & 0 \\ a_1 & a_2 & a_3-b_3 & 0 & 0 \\ a_1 & a_2 & a_3 & a_4 & a_n-b_n \end{bmatrix}$$

$$C = \begin{bmatrix} a_1 & 0 & 0 & 0 & 0 \\ a_1 & a_2 & 0 & 0 & 0 \\ a_1 & a_2 & a_3 & 0 & 0 \\ a_1 & a_2 & a_3 & a_4 & 0 \\ a_1 & a_2 & a_3 & a_4 & a_n \end{bmatrix}$$

Put

$$a(z) = v(x) + w(x) .$$

Now the equation (2.6) becomes

$$\begin{aligned} z''(x) + D(B+C)z(x) &= -2f(x)D\eta \\ z(0) &= w(0) \\ z'(0) &= Dw(0) \end{aligned} \tag{2.7}$$

where

$$D = \begin{bmatrix} b_1 & 0 & 0 & 0 & 0 \\ b_1 & b_2 & 0 & 0 & 0 \\ b_1 & b_2 & b_3 & 0 & 0 \\ b_1 & b_2 & b_3 & b_{4..} & b_n \end{bmatrix}$$

This is an initial value problem which can be solved provided we know the initial value w(0).

3. Inverse Laplace Transform

We utilize the Inverse Laplace Transform method to find the initial value. Now the equation (2.6) can be written as

$$r'(x) - Er(x) = F(x), \tag{3.1}$$

where

$$r(x) = \begin{bmatrix} v(t) \\ w(x) \end{bmatrix}$$

$$E = \begin{bmatrix} -B & -C \\ C & B \end{bmatrix}$$

and

$$\Gamma(x) = f(x) \begin{bmatrix} \eta \\ -\eta \end{bmatrix}$$

Integrating (3.1),

$$e^{E(x)} r(x) \Big|_o^a = \int_o^a e^{E(x)} F(x)\,dx,$$

write

$$\ell(e^{E(x)}) = [Is-E]^{-1}$$

where I is an n x n unit matrix. Then

$$e^{E(x)} = \ell^{-1}\{[Is-E]^{-1}\} .$$

Writing in detail the eqn. (3.2),

$$e^{E(a)} \begin{bmatrix} v(a) \\ w(a) \end{bmatrix} - e^{E(0)} \begin{bmatrix} v(0) \\ w(0) \end{bmatrix} = \int_o^a e^{E(x)} F(x)\,dx .$$

Here we have a system of simultaneous equations involving two unknown v(z) and w(0). So we can find w(0) explicitly once we determine exp (E(x)). Thus we can solve for z(x) and hence u(x) from the eqn. (2.2).

4. An Example

The integral equation

$$u(t) - g(t) + \frac{1}{2} \int_o^1 e^{|t-y|} u(y)dy \ , \ 0 \leqslant t \leqslant 1 \cdot \qquad (4.1)$$

is encountered in a one dimensional transport process. The function u(t) is the cource function or the total rate of scattering of energy to the left or to the right at the point t in a rod extending from t - 0 to t = 1. The function g(t) represents the rate of primary emission of energy at t. Suppose that

$$g(t) = \frac{1}{2}[e^{-t} + e^{-(1-t)}]$$

so that the solution of the equation (4.1) is

$$u(t) = 1 .$$

The equation (3.1) becomes for this case

$$r'(t) - Er(t) = (e^{-t} - e^{-(1-t)}) \begin{bmatrix} 1 \\ -1 \end{bmatrix} , \qquad (4.2)$$

where

$$E = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

now

$$\exp(E(t) = \ell^{-1}\{[Is-E]^{-1}\}$$

$$= \begin{bmatrix} 1+\frac{1}{2}t & -\frac{1}{2}t \\ \frac{1}{2}t & 1-\frac{1}{2}t \end{bmatrix} .$$

Integrating (4.2), we get

$$\begin{bmatrix} 1+\frac{1}{2}t & -\frac{1}{2}t \\ \frac{1}{2}t & 1-\frac{1}{2}t \end{bmatrix} \begin{bmatrix} v(t) \\ w(t) \end{bmatrix} \Bigg|_0^1 = \int_0^1 \begin{bmatrix} 1+\frac{1}{2}t & -\frac{1}{2}t \\ \frac{1}{2}t & 1-\frac{1}{2}t \end{bmatrix} \begin{bmatrix} e^{-t}-e^{-(1-t)} \\ -e^{-t}+e^{-(1-t)} \end{bmatrix} dt$$

This gives on simplification

$$\frac{3}{2}v(1) = \frac{3}{2}(1-e^{-1}) ,$$

$$\frac{1}{2}V(1) = w(0) + \frac{1}{2}(e^{-1}-1) .$$

Therefore,

$$w(0) = 1-e^{-1} .$$

The equation (2.7) takes the form

$$z''(t) = -e^{-t}-e(^{-(1-t)} ,$$

$$z(0) = z'(0) = w(0) = 1-e^{-1} .$$

This gives

$$z(t) = 2 - e^{-t}-e^{-(1-t)} .$$

With the help equation (2.2), we write

$$u(t) = e^{-t} + e^{-(1-t)} + z(t).$$

$$u(t) = 1 ,$$

which is the solution to the integral equation.

5. Acknowledgements

The author expresses his gratitude to Professor R. E. Bellman for his generous support and counsel. Thanks are due to Professors R. E. Kalaba and R. M. Rao for helpful discussions.

References

Bellman, R. E. 1967. Introduction to Control Theory, Academic Press, New York.

Mohanty, N. C., 1971. Differential Approximation and Linear Integral Equation, Fifth Princeton Conference on Information Science and Systems.

Ogata, K., 1967. State-Space Analysis of Control Systems. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Vantrees, H. L., 1968. Detection, Estimation, and Modulation Theory, Part I, John Wiley and Sons, Inc., New York.

TRANSITION PARTITIONS FOR STATE ASSIGNMENT IN SEQUENTIAL MACHINES*

A. JAI and J. W. GAULT
Dept. of Electrical Engineering, N. C. State University
Raleigh, North Carolina 27607

ABSTRACT

A new set of partitions on the states of a sequential machine are defined in terms of transition behavior. This approach gives a unified set of partitions which are independent of the eventual flip-flop to be used and require only three generation rules. With the knowledge of the FF type, candidate partitions are selected from the transition set and scored. The new partitions generate all useful members and their relation to the π, ρ, τ partitions previously used is given.

1. INTRODUCTION

The assignment of binary codes to the internal states so as to reduce the complexity of the circuit realization is a major problem in the design of sequential machines. Numerous approaches involving different assumption have been undertaken towards solving this problem.

The earlier approaches to the state assignment problem involved the idea of reduced variable dependence in next state equations to simplify the circuit realization. These methods applied mainly to sequential machines using delay type memory elements.

In practice, however, sequential machines are realized using a variety of flip-flop memory elements. The excitation equations of flip-flops do not have the same variable dependencies as delay memory elements. Hence, the concept of reduced variable dependence in next state equations is inadequate [2,4].

Recent work considering the use of different types of flip-flops utilizes the transition behaviour of the flip-flops and the structural properties of the flow table of the sequential machine [1,2]. These methods rely on the identification of a set of useful properties of excitation functions as a means of predicting a relatively simple set of excitation equations [3].

Partitions are a useful tool for systematic state assignment procedures. Usefulness of partitions lies in the relative computational ease with which sets of partitions with useful properties can be identified and generated. The basic thrust has been to generate a set of potentially useful partitions (PUPs) from the state table, score these PUPs based on their possessing the desirable properties, and then make a selection of those with the "best" scores [1,2].

Although Curtis [2] and Cotter [1] derived the set of PUPs using different techniques, their work evolved as extension from delay memory to eventually dealing with J-K flip-flops. This approach seems to have led both methods to overlook some useful partitions when J-K implementations are considered. The objective here, therefore, is to present a unified method (independent

---

*This work was supported by the Army Research Office, Durham, NC

of FF type) of generating a set of candidate partitions. Once the FF type which is to be used in an implementation is determined then a complete set of PUP's can readily be selected from this set of candidates.

The PUPs generated here are derived using the transition behaviour which defines various flip-flop types. The partitions which are defined are shown to generate a more complete set of PUPs which includes all of those partition types previously defined by others.

The development is organized in the following way. The desirable properties of excitation equations under "good" state assignments are given. Partitions are then related to state assignments and the desirable properties of "good" equations. At this point the primary idea of this paper is given, namely the generation of partitions which reflect the nature of transition behaviour in a state table independent of FF type. For a particular FF type the desirable properties of an equation can be related to specific members of the transition partition set. The correspondence is given for all FF types. This then gives a set of partitions (selected from the transition partitions) which may be scored to give a "good" state assignment.

2. TRANSITION PARTITIONS

Equation properties

In the state assignment procedures developed by Cotter [1] and Curtis [2], the simplification of next-state equations was considered on a per column basis. If there are n state variables and a flow table of m columns (m input states), the general form of the next-state equations will be

$$Y_i = f_{i1}(\overline{Y})\ I_1 + f_{i2}(\overline{Y})\ I_2 + \cdots + f_{im}(\overline{Y})\ I_m \qquad (1)$$

$$i = 1,2,\cdots,n$$

where $\overline{Y} = y_1, y_2, \cdots, y_n$ is the vector of the present-state variables; $I_1, I_2, \cdots, I_m$ are the input states; $f_{i1}, f_{i2}, \cdots, f_{im}$ for all i are functions of present state variables alone; Y_i is the next-state function for the i^{th} memory element. For trigger T, and J-K flip-flops Y_i will be denoted by T_i, J_i, and K_i, respectively, and will represent the input excitation functions for these flip-flops.

Using this m column formulation of the next state equations, the desirable characteristics of a state assignment, independent of the choice of the memory element, are those which yield simple excitation equations. This basically calls for making the coefficients f_{ik} simple or related to other coefficients. The properties considered here are 1) $f_{ik}=0$ or 1, 2) $f_{ik}=y_i$ or y_i', 3) $f_{ik}=f_{i\ell}$ when I_k and I_ℓ are adjacent columns [3].

In the excitation table of any sequential circuit these characteristics may be visualized as

1) A next-state excitation of all '0' or all '1' entries for Y_i for any column,
2) A present-state code for y_i equal to or complement of the next-state excitation for Y_i for any column.
3) Next-state excitation for Y_i in column k equal to the next-state excitation for Y_i in column ℓ when I_k, I_ℓ are adjacent input columns (i.e., codes distance 1 apart).

It is well known that partitions may be used to define assignments for state variables. More important there is the fact that partitions may be used to define assignments of variables which have one of the useful properties. These partitions are called PUPs.

For the machine (A) of figure 2.1a it is possible to define the partitions of fig. 2.1c. These partitions are elements of the PUP set and form the assignment given by 2.1b. This selection of partitions can be seen to give several of the useful properties listed before.

NS

			PS			I_1			I_2			
PS	I_1	I_2	y_1	y_2	y_3	Y_1	Y_2	Y_3	Y_1	Y_2	Y_3	
1	2	3	1	0	0	0	0	0	0	1	0	$\theta_i=\{A,B\}$
2	4	5	0	0	0	0	0	1	1	1	1	$\theta_1=\{234,15\}$
3	3	2	0	1	0	0	1	0	0	0	0	$\theta_2=\{124,35\}$
4	2	1	0	0	1	0	0	0	1	0	0	$\theta_3=\{123,45\}$
5	3	1	1	1	1	0	1	0	1	0	0	

(a) State Table Machine A (b) Excitation table for delay memory element (c) Partitions

Figure 2.1 Sequential Machine A

Column Y_1 under I_1 is an all '0' column; PS column for y_1 is equal to the next-state column for Y_2 under I_1; and next state column for Y_3 under I_1 is equal to the next-state column for Y_3 under I_2.

Transition Partitions

The objective then is to generate a set of partitions which define a state assignment which results in simplified FF excitation equations. The approach taken here is to develop, from an uncoded state table, a set of partitions which reflect potential present state/next state transition behaviour. The selection of a particular FF type then imposes certain excitation requirements based on the desired state transitions. State assignments with desirable properties can be derived from the transition partitions and knowledge of the FF type.

Irrespective of the type of memory element used the excitation table for any sequential circuit has only four types of transitions from y_i to Y_i, namely, 0→0, 0→1, 1→0, and 1→1. Transition partitions describe these transitions.

The transition partitions of interest are of three types, all of which have a complementary partition set also of interest. For each input I_k a set of partitions will be generated to describe the condition that no 0→0 transition exists. Its complement is no 1→1 transition. Similarly a partition set is required for the condition that no 0→1 transition exists. Its complement is no 1→0 transition. The third partition set defines the condition between adjacent columns, namely, that if a 0→0 (0→1) transition exists for state s_i and input I_k, then a 0→0 (0→1) transition must exist for state s_i and input I_ℓ, where k and ℓ are adjacent columns. This partition set also has a useful complement with transitions 1→0 (1→1).

Table 2.1 summarizes these three sets of Transition Partitions and their conditions.

Transition Partition Set	Transition $y_i \to Y_i$	
Set 1	no 0 → 0	for input I_k
Set 1	no 1 → 1	" " "
Set 2	no 0 → 1	" " "
Set 2	no 1 → 0	" " "
Set 3	0 → 0 (0 → 1)	exist simultaneously for state s_i and adjacent inputs I_k and I_ℓ
Set 3	1 → 1 (1 → 0)	

Table 2.1 Transition Partition Sets

The rules for generating the partitions sets given above are reasonably straightforward. Beginning with an uncoded state table, the generation rules are:

Set 1: No 0→0 transition (for I_k)

Rule 1: If s_i ε of block A (0-block) then its successor and predecessor states under I_k must be in block B (1-block).

Set 2: No 0→1 transition (for I_k)

Rule 2: If s_i ε of block A (0-block) then its successor states under I_k must also be in block A (0-block).

Set 3: 0→0 (0→1) transitions (for I_k, I_ℓ)

Rule 3: If s_i ε of block A (0-block) then its successor states under I_k and I_ℓ must be in a common block.

The complementary sets are obtained by simply complementing every partition of the respective sets. If $\theta_i=\{A,B\}$, then $\overline{\theta}_i=\{B,A\}$. The reason for this is that if blocks A and B are interchanged then all 0→0 transition become 1→1 transitions, and similarly all 0→1 transitions become 1→0 transitions.

Partition sets for machine A (Fig. 2.1) derived using these rules are listed below in Fig. 2.2. The complementary sets are implied.

PUP Selection

This set of candidates partitions was derived independent of any knowledge of the FF type to be used. Now, with the FF type selected, a set of PUPs may be selected from the list of candidates. A candidate partition is a PUP iff for the particular FF type in question that partition will result in an assignment and hence an excitation equation with one of the three useful properties.

All potentially useful partitions (PUPs) having the above mentioned useful properties using any type of flip-flop may now be derived from the three sets of transition partitions.

The input excitation functions for the various types of flip-flops are derived using the excitation characteristics of the respective flip-flops shown in Table 2.2.

Input I_1		Input I_2		Inputs I_1, I_2
Set 1	Set 2	Set 1	Set 2	Set 3
{1,2345}	{3,1245}	{1,2345}	{1235,4}	{1,2345}
{2,1345}	{35,124}	{2,1345}		{2,1345}
{4,1235}	{24,135}	{3,1245}		{4,1235}
{5,1234}	{234,15}	{4,1235}		{5,1234}
{14,235}	{124,35}	{5,1234}		{23,145}
{15,234}	{1234,5}	{12,345}		{45,123}
{25,134}	{2345,1}	{24,135}		{123,45}
{45,123}		{34,125}		
{145,23}		{35,124}		
		{45,123}		
		{345,12}		

Figure 2.2 Transition partitions of machine A

Transition	Excitation required			
y_i Y_i	Y_i	T_i	J_i	K_i
0 → 0	0	0	0	-
0 → 1	1	1	1	-
0 → -	-	-	-	-
1 → 0	0	1	-	1
1 → 1	1	0	-	0
1 → -	-	-	-	-

Table 2.2 Excitation characteristics for Delay, T, and J-K flip-flops.

For a D-type flip-flop from Table 2.2 under Y_i we observe that if there are no 0→1 and no 1→1 transitions, then the input excitations would be either 0,- (the don't cares (-) may be assigned a value 0).

Thus, for a D flip-flop property 1 is equivalent to no 0→1 (Set 2) and no 1→1 (Set 1) transitions for a particular input column.

Thus partitions common to Set 2 and $\overline{\text{Set 1}}$ (i.e., {1,2345}, {5,1234},{15,234}/I_1) will give all the partitions (PUPs) that generate an all '0' excitation (property 1). Similar argument may be presented for partition that generate an all '1' excitation.

Again, referring to Table 2.2, $y_i=Y_i$ for transitions 0→0 and 1→1. So if the remaining transitions are absent, i.e., no 0→1 and no 1→0, then property 2 will be satisfied. Thus partitions common to Set 2 and $\overline{\text{Set 2}}$ satisfy property 2 (i.e., {124,35}/I_1).

To satisfy property 3 for D-flip-flops all transitions for column k must be the same as their corresponding transitions for column ℓ. All four types of transitions specify Y_i. Set 3 satisfies the case for 0→0 (0→1) and $\overline{\text{Set 3}}$ satisfies the case for 1→1 (1→0). Thus partitions common to Set 3 and $\overline{\text{Set 3}}$ (i.e., {123,45}) satisfy all four types of transitions and, hence, satisfy property 3 for D-flip-flops. All the useful properties formulated as transition behaviour for D-flip-flop are summarized below.

1. $f_{ik}(Y)=1$ no 0→0 and no 1→0 ($\overline{\rho}$) Set 1 Ω $\overline{\text{Set 2}}$
   $f_{ik}(Y)=0$ no 1→1 and no 0→1 (ρ) $\overline{\text{Set 1}}$ Ω Set 2
2. $f_{ik}(Y)=y_i$ no 0→1 and no 1→0 (π) Set 2 Ω $\overline{\text{Set 2}}$
   $f_{ik}(Y)=y_i'$ no 0→0 and no 1→1 (τ) Set 1 Ω $\overline{\text{Set 1}}$
3. $f_{ik}(Y)=f_{i\ell}(Y)$ 0→0 (0→1) and 1→1 (1→0) (α) Set 3 Ω $\overline{\text{Set 3}}$
   for s_i under I_k and I_ℓ

The π, ρ, $\overline{\rho}$, τ, and α nomenclature correspond to definitions given by Cotter [1]. Similar correspondence may be drawn with conditions given by Curtis [2]. Similar derivations may be given to develop sets of candidate partitions for T, J-K flip-flops. Table 2.3 summarizes these results.

Flip-flop type	Property				
	1		2		3
	$f_{ik}(Y)=0$	$f_{ik}(Y)=1$	$f_{ik}(Y)=y_i$	$f_{ik}(Y)=y_i'$	$f_{ik}(Y)=f_{i\ell}(Y)$
D	$\overline{S\ 1}$ Ω S 2	S 1 Ω $\overline{S\ 2}$	S 2 Ω $\overline{S\ 2}$	S 1 Ω $\overline{S\ 1}$	S 3 Ω $\overline{S\ 3}$
T	S 2 Ω $\overline{S\ 2}$	S 1 Ω $\overline{S\ 1}$	$\overline{S\ 1}$ Ω S 2	S 1 Ω $\overline{S\ 2}$	S 3 Ω $\overline{S\ 3}$
J-K J_i	Set 2	Set 1	Set 2	Set 1	Set 3
J-K K_i	$\overline{\text{Set 2}}$	$\overline{\text{Set 1}}$	$\overline{\text{Set 1}}$	$\overline{\text{Set 2}}$	$\overline{\text{Set 3}}$

Table 2.3 Partition sets for Delay, T, and J-K flip-flops

The partition generated for the J-K case are more extensive (but include) the work of Cotter [1] and Curtis [2]. Implicit in their work was the assumption that the useful properties of the state assignment be valid for both the J-input and the K-input excitations simultaneously. There is no such assumption here.

3. SUMMARY

A partition theory approach to the state assignment problem for synchronous sequential machines has been given. This approach unifies the derivation of candidate partitions and provides three straightforward derivation rules which are common for all flip-flops. The final PUP selection is accomplished by selecting from the candidate set based on the behaviour rules of particular FF types. The set of partitions derived includes and extends previous work.

Once a set of PUPs has been derived, scoring and final state code selection may proceed as before.

4. REFERENCES

1) Cotter, D. A., "On Designing Minimal Cost Synchronous Sequential Circuits", Ph.D. dissertation, North Carolina State University, 1970.

2) Curtis, H. A., "Systematic Procedures for Realizing Synchronous Sequential Machines Using Flip-Flop Memory", IEEE Trans. on Computers, Part I, December 1969, Part II, January 1970.

3) Dolotta, T. A. and E. J. McCluskey, "The Coding of Internal States of Sequential Circuits", IEEE Trans. on Computers, EC-13, October 1964.

4) Harlow, C. and C. L. Coates, "On the Structure of Realizations Using Flip-Flop Memory Circuits", Information and Control, 10:159-174.

DESIGN EQUATION GENERATION FOR ASYNCHRONOUS SEQUENTIAL CIRCUITS USING PARTITION THEORY

DWIGHT H. SAWIN III
U.S. Army Electronics Command
Fort Monmouth, New Jersey

GARY K. MAKI
University of Idaho
Moscow, Idaho

Abstract - A critical step in realizing an asynchronous sequential circuit is the generation of the design equations from a flow table and its corresponding STT state assignment. This paper presents a generation method employing partition algebra and a fundamental relationship between Tracey's π-partitions (1), and the design equations. Partition algebra allows the procedure to be used for circuit realization employing memory elements as well as gates. This is an improvement over previously derived procedures (2,3). Furthermore the procedure is easier to computerize than other methods (4), because it is algorithmic and fundamental in nature.

I. INTRODUCTION

Generation of the next state and output equations is a key step in the synthesis procedure for asynchronous sequential circuits, because without it the physical realization of the circuit cannot be obtained. It is difficult when large, practical circuits, making procedures using transition tables and Karnaugh maps, etc., unmanageable, are to be analyzed or realized. At this point all the "theory" describing the circuit is lost and a trial and error process begins. By using partition algebra and the direct relationship which exists between Tracey's π-partitions, the design equations and the state assignment, the problem can be reduced to a procedure which is applicable to large, practical circuits.

Previous attempts at realizing asynchronous sequential machines with a minimum amount of hardware have turned to the computer to do the volume of computations. Smith, et. al., (4) developed a program based on a derivation technique presented by Maki, et. al., (2). This method consists of a rather large amount of unwieldy computation, e.g., their algorithm for handling unspecified or don't care states is difficult to execute even with a computer. Tan (3) developed a technique employing partitions for non-memory element realizations. While the rudiments of his procedure are used here the theory is developed in greater depth, placing the state assignment partitions, design equations and weighting techniques for state assignment selection (5,6) in context with each other. The method applies to realizations using memory elements as well as logic gates. Other techniques apply only to gate realizations, which are not often used in practice.

Circuits considered here are fundamental mode asynchronous sequential circuits with single-transition-time (STT) state assignments (1,7). Their operation can be described by means of a flow table such as the one shown in Figure 1. Also shown is a valid internal state assignment.

y_1	y_2	y_3	y_4		I_1	I_2	I_3
1	1	1	0	A	B	Ⓐ/1	F
1	1	0	1	B	Ⓑ/0	A	Ⓑ/0
0	1	0	1	C	B	F	B
0	0	1	0	D	Ⓓ/1	E	F
1	0	0	0	E	D	Ⓔ/1	B
0	1	1	1	F	D	Ⓕ/0	Ⓕ/1

Figure 1. Machine A

Figure 2 shows a generalized model for memory elements. For gate realizations $l=1$ and y_i simply follows the input, i.e. the memory element performs the identity function. (y_i' is produced with an inverter.) For SR flip-flops, $l=2$ and y_i and y_i' are available with ${}^1Y_i$ and ${}^2Y_i$ corresponding to S_i and R_i. Likewise the model may be used for JK flip-flops or other devices.

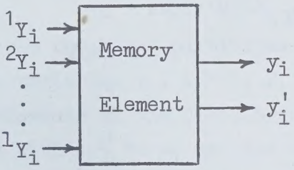


Figure 2. Memory Element Model for Variable Y_i

The next state equations for a flow table can be described on a per-column basis. If there are n internal state variables, a flow table of m columns, and a realization employing memory elements, with l inputs, the general form of the next state expressions is

$$
\begin{aligned}
{}^1Y_i &= \sum_{p=1}^{m} {}^1f_i^p(y_1, y_2, \ldots, y_n)I_p \\
{}^2Y_i &= \sum_{p=1}^{m} {}^2f_i^p(y_1, y_2, \ldots, y_n)I_p \\
&\vdots \\
{}^lY_i &= \sum_{p=1}^{m} {}^lf_i^p(y_1, y_2, \ldots, y_n)I_p .
\end{aligned}
$$

where $y_1, y_2, \ldots, y_n$ are the present state variables, ${}^1Y_i, {}^2Y_i, \ldots, {}^lY_i$ are the next state inputs to the memory element; $I_1, I_2, \ldots, I_m$ are the input states; and ${}^1f_i^1, {}^1f_i^2, \ldots, {}^lf_n^m$ are functions of internal state variables only. The output equations also take the same form (4).

II. NEXT STATE VARIABLE GENERATION

Design equation derivation now reverts to solving for the coefficient terms, ${}^lf_n^m$. First the action of the memory device selected is defined and then a vast majority of the coefficient terms are determined by inspection.

The number of coefficient terms obtainable by inspection depends on how well the selected state assignment is suited for the flow table. If the selection process (5,6) is performed well all the coefficient terms may be obtained by inspection.

A more detailed procedure is used to solve for the remaining coefficients. The algorithm also allows for computation of terms which may be shared between the memory device inputs (a form of the classical multiple output problem) and thus maximum hardware simplicity. It should be emphasized that the "price" for maximum simplicity in hardware is computation time. Also unspecified or don't care states simply fall into place during the algorithm execution requiring no special attention in contrast to other algorithms (2).

A convenient method for describing memory element operation uses the circuit action symbols described by Marcus (9). These symbols are defined as follows for the event $y \rightarrow Y$.

Event	Symbol
$0 \rightarrow 1$	Φ
$1 \rightarrow 0$	Θ
$1 \rightarrow 1$	1
$0 \rightarrow 0$	0

Use of the symbols makes design easier since they can be used for implementation with SR or JK flip-flops or gates. (A more complex set of rules is required for more complicated devices such as SRCD flip-flops, etc.) The rules to obtain excitation expressions for logic gates, in a sum-of-products form, consist of accounting for each Φ or 1, but no Θ or 0 in the expression. For SR flip-flops each Φ but no Θ or 0 must be accounted for in the expression representing the S input. For the R input each Θ but no Φ or 1 must be accounted for. Similar rules are given for other devices by Marcus (9).

Definition 1: A pair of states consisting of one unstable state and the stable state of a k-set is called a transition pair.

Definition 2: A partition Π on a set S is a collection of subsets of S such that their pairwise intersection is the null set. The disjoint subsets are called blocks of Π. If the union of these subsets is S, the partition is completely specified; otherwise, the partition is incompletely specified. Elements of S that do not appear in Π are called unspecified or optional with respect to that partition.

The following theorem, first shown by Tracey (1), gives the necessary and sufficient conditions for minimum variable STT state assignments, in terms of partitions.

Theorem (Tracey): A state assignment allotting one y-state (n-tuple) per state can be used for realization of STT flow tables without critical races if, and only if, the following conditions hold for every transition pair $\{S_i,S_j\}$, where S_j is a stable state:

a) if $\{S_m,S_n\}$ is another transition pair in the same column, $j \neq n$, then at least one partition partitions the pair $\{S_i,S_j\}$ and the pair $\{S_m,S_n\}$ into separate blocks and

b) if S_k is a stable state in the same column, $j \neq k$, then at least one partition partitions the pair $\{S_i,S_j\}$ and the state S_k into separate blocks and

c) for $i \neq j$, S_i and S_j are in separate blocks of at least one partition.

The partition list, i.e., those π-partitions fulfilling the Tracey conditions, for Machine A is:

$\pi_1 = \{\overline{A,B};\ \overline{D,E}\}$ $\pi_4 = \{\overline{B,C};\ \overline{D,F}\}$ $\pi_7 = \{\overline{A,F};\ \overline{B,E}\}$

$\pi_2 = \{\overline{A,B};\ \overline{D,F}\}$ $\pi_5 = \{\overline{C,F};\ \overline{A,B}\}$ $\pi_8 = \{\overline{A,F};\ \overline{B,E}\}$

$\pi_3 = \{\overline{B,C};\ \overline{D,E}\}$ $\pi_6 = \{\overline{C,F};\ \overline{D,E}\}$ $\pi_9 = \{\overline{D,F};\ \overline{B,C}\}$

$\pi_{10} = \{\overline{D,F};\ \overline{B,E}\}$.

Definition 3: The ordered two-block partitions τ_i, $i = 1, 2, \ldots, n$ induced by internal state variables y_i, $i = 1, 2, \ldots, n$ are called the set of τ-partitions. The elements of the first block of τ_i are those internal states that are encoded with a 1 by y_i and the elements of the second block are those states encoded with a 0 by y_i.

In Figure 1, the τ-partitions are: $\tau_1 = \{\overline{A,B,E};\ \overline{C,D,F}\}$, $\tau_2 = \{\overline{A,B,C,F};\ \overline{D,E}\}$, $\tau_3 = \{\overline{A,D,F};\ \overline{B,C,E}\}$ and $\tau_4 = \{\overline{B,C,F};\ \overline{A,D,E}\}$.

Valid STT state assignments are obtained by generating a set of τ-partitions such that each π-partition is contained at least once in some τ-partition. Methods for deriving such a set of τ-partitions are discussed by Tracey (1) and Unger (7).

Definition 4: Transition pair partition, $^{l}P^{p}_{i}(wz)\{A;B\}$, partitions transition pair wz, having one state with circuit action event A, from transition pairs having states with circuit action event B for $^{l}Y_i$ under I_p.

Throughout the basic explanation of the procedure for deriving the next state equations, Machine A of Figure 1, with a logic gate realization will be used for illustration purposes. Once the method is understood an application for SR flip-flop design will be given. For logic gates $^{l}P^{p}_{i}(wz)$ partitions Φ and 1 events from Θ and 0 circuit action events. Thus A would contain a transition pair with a Φ or 1 event and B contains transition pairs with Θ or 0 events. For gate realizations $l = 1$ and hence is omitted for simplicity. Two transition pair partitions for Y_1 under I_1 of Machine A are $P^1_1(AB) = \{\overline{A,B};\ \overline{D,E,F}\}$ and $P^1_1(BC) = \{\overline{B,C};\ \overline{D,E,F}\}$. By composing $P^1_1(AB)$ and $P^1_1(BC)$ into one partition all the states under I_1 which have a next state entry of Φ or 1 for Y_1 are depicted.

Definition 5: Next state partition, ${}^{1}\eta_i^{p}\{A;B\}$, partitions all those transition pairs where circuit action event A occurs from those where circuit action event B occurs for ${}^{1}Y_i$ under input I_p.

The first step in determining the coefficient terms, ${}^{1}f_i^{p}$, is to form the next state partitions. The next state partitions for I_1 of the flow table in Figure 1 are:

$$\eta_1^1 = \{\overline{A,B,C};\ \overline{D,E,F}\} \qquad \eta_3^1 = \{\overline{D,E,F};\ \overline{A,B,C}\}$$
$$\eta_2^1 = \{\overline{A,B,C};\ \overline{D,E,F}\} \qquad \eta_4^1 = \{\overline{A,B,C};\ \overline{D,E,F}\}.$$

Employing the following three theorems presented by Maki, Sawin and Jeng (6), a large number of the coefficient terms can now be identified by inspection.

Theorem 1: If all the states of ${}^{1}\eta_i^{p}$ appear in only one block then ${}^{1}f_i^{p} = 1$ or 0.

Theorem 2: If ${}^{1}\eta_i^{p} \leq \tau_j$ then ${}^{1}f_i^{p} = y_j$ or y_j'.

Theorem 3: If ${}^{1}\eta_i^{p} \leq {}^{1}\eta_j^{p}$, then ${}^{1}f_i^{p} = {}^{1}f_j^{p}$ or ${}^{1}f_j^{p\,\prime}$.

y_3y_4 \ y_1y_2	00	01	11	10
00				E
01		C	B	
11		F		
10	D		A	

Figure 3. State Assignment

Definition 6: The product of partitions $\tau_1, \tau_2, \ldots, \tau_k$, $\tau_1 \cdot \tau_2 \cdot \cdots \cdot \tau_k$, is a two-block partition where only those states appearing in the first block of $\tau_1, \tau_2, \ldots, \tau_k$ appear in the first block of the partition. All other states appear in the second block.

Consider $\tau_1 = \{\overline{A,B,E};\ \overline{C,D,F}\}$ and $\tau_2 = \{\overline{A,B,C,F};\ \overline{D,E}\}$, then $\tau_1' \cdot \tau_2 = \{\overline{C,F};\ \overline{A,B,}\ \overline{D,E}\}$. One of the largest p-subcubes that contains states CF in Figure 3 is mapped onto $y_1'y_2$. Also $\tau_1' \cdot \tau_2 = \eta_4^2$, thus $y_1'y_2$ partitions the transition pair CF from all other states. Note that $\tau_1' \cdot \tau_4 = \eta_4^2$; hence $y_1'y_4$ is another possible term to map onto η_4^2. Generation of the next state coefficients, f_i^p, is simply a problem of finding the sum-of-products expression which each next state partition can be mapped onto.

Each transition pair partition can be expressed as a union of π-partitions. For example, $P_4^2(CF) = \pi_5 \cup \pi_6$, where $\pi_5 = \{\overline{C,F};\ \overline{A,B}\}$ and $\pi_6 = \{\overline{C,F};\ \overline{D,E}\}$. In general, $P_i^p(wz)$ is a union of those π-partitions from I_p that partition wz. The τ-partitions that cover these π-partitions are used to form the product partition. For instance, $\pi_5 \leq \tau_1'$ and $\pi_6 \leq \tau_2$, then $\tau_1' \cdot \tau_2 = \{\overline{C,F};\ \overline{A,B,D,E}\}$, which partitions those states in the transition path of CF where $Y_4 = 1$ from those states for which $Y_4 = 0$. Furthermore,

$P_4^2(CF) \leq \tau_1' \cdot \tau_2$ and $P_4^2(CF)$ maps onto the product expression $y_1' y_2$ and represents the 1's in the transition path of CF. $\tau_1' \cdot \tau_4$ also would suffice. The following theorem establishes the basis for the above procedure.

Theorem 4: For ${}^1P_i^p(wz)$ containing π-partitions π_i, i=1, 2, . . . , s, there exists τ_j such that $\pi_i \leq \tau_j$. If the states wz are in the first block of τ_j then ${}^1P_i^p(wz) \leq \tau_1 \cdot \tau_2 \cdot . . . \cdot \tau_s$ and the product expression for ${}^1Y_i$ in the transition path of wz is $y_1 \cdot y_2 \cdot . . . \cdot y_s$.

Corollary 1: If ${}^1P_i^p(wz) \leq \tau_1 \cdot \tau_2 \cdot \tau_j' . . . \cdot \tau_s$, where τ_j' denotes reversing the order of τ_j's blocks, then the product expression for ${}^1Y_i$ in the transition path of wz is $y_1 \cdot y_2 \cdot . . . \cdot y_j' \cdot . . . \cdot y_s$.

Theorem 4 and Corollary 1 follow directly from Tan's (3) Lemma 2, even through the nomenclature is different and these results are much more general in nature. They are significant because they show that finding a product expression for any input to a memory element is simply a covering problem. (Note that an MSP expression is obtained but reversing the order of the blocks of the partitions yields an MPS.) The essence of the problem is to determine the τ-partitions which cover the π-partitions used to form the transition pair partition. A table can be made and used to find all irredundant expressions. Figure 4 shows a table of $P_4^2(CF)$ where X' denotes that CF is coded 0 by y_i. Covering the table may be accomplished by many techniques. One such method is Petrick's method which, for the table in Figure 4, yields:

	π_5	π_6
τ_1	X'	
τ_2		X
τ_3		
τ_4		X

Figure 4. τ-partition Containments for $P_4^2(CF)$.

$$P_4^2(CF) \leq \tau_1' \cdot \tau_2 \text{ or } \tau_1' \cdot \tau_4 \text{ and}$$

$$f_4^2 = y_1' y_2 \text{ or } y_1' y_4.$$

Petrick's method, as applied above, yields all minimum irredundant terms. Note however that another product for the p-subcube describing CF is $y_1' y_2 y_4$ (2). Thus Petrick's method yields a minimum description for the p-subcubes but an unexcited variable may be ANDed with these results. While not logically minimum, these larger terms also accurately describe the transition path.

In general, a next state partition is the union of transition pair partitions for ${}^1Y_i$ under I_p. For $\eta_1^1 = \{\overline{A,B,C}; \overline{D,E,F}\}$, $P_1^1(AB) = \{\overline{A,B}; \overline{D,E,F}\}$ and $P_1^1(BC) = \{\overline{B,C}; \overline{D,E,F}\}$. Thus $\eta_1^1 = P_1^1(AB) \cup P_1^1(BC)$. Since $P_1^1(AB) \leq \tau_1'$.

τ_2, $P_1^1(BC) \leq \tau_2 \cdot \tau_3'$ and $\tau_3' \cdot \tau_4$, coefficient $f_1^1 = y_1'y_2 + y_2y_3'$ or $y_1'y_2 + y_3'y_4$. The logic expression for f_i^p is the sum of the product terms that result from each of the $P_i^p(wz)$ partitions that compose the next state partition.

Following is an algorithmic statement of the procedure to find each coefficient term:

1. List the π-partitions and the τ-partitions.
2. Select the type of realization to be derived.
3. Form the next state partitions.
4. Apply Theorems 1, 2, and 3 to identify as many coefficient terms as possible. Note the next state partitions that cannot be covered.
5. For each undefined next state partition:
    a. Find the transition pair partition(s).
    b. Find the product expression for each transition pair partition by Petrick's method or some other covering technique.
    c. OR each product expression found for each ${}^1P_i^p(wz)$ of ${}^1\eta_i^p$ to find ${}^1f_i^p$.

The next state equations have now been independently derived. After the input states have been coded, further minimization is achieved by considering the equations as a multiple output logic design problem. To arrive at a minimal set of equations all possible coefficient terms must be considered to maximize the sharing of gates. Consideration must be given to sharing logic between both next state variables and input states.

Removal of static hazards is a vital concern for logic designers. Using two level logic synthesis all transitions are static hazard free if each implicant covers only transition pairs (4). Hence static hazards may only be caused by input state changes which correspond to horizontal transitions in the flow table. Under the restriction that only one input variable is allowed to change at a time, "horizontal" hazards may be eliminated by using a modification of McCluskey's hazard free synthesis technique (8).

Most flow tables are single-output-change (SOC) tables. In SOC tables, all states of a transition path have the same output, namely that of the stable state of the k-set (7). Obtaining the output equations is a simple extension of the next state equations (2).

III. EXAMPLE OF SR FLIP-FLOP DERIVATION

To illustrate the versatility of this method, the next state equations for I_1 of Machine A, employing an SR flip-flop realization are now derived. Since the π-partitions and τ-partitions have already been listed and the realization type has been selected, this example begins with step 3 of the algorithm.

Step 3: The next state partitions ${}^1\eta_i^p$ and ${}^2\eta_i^p$ correspond to the S and R inputs respectively. Thus ${}^1\eta_i^p({}^2\eta_i^p)$ partitions transition pairs with Φ (θ) events from transition pairs with θ or 0 (Φ or 1) events. Hence,

$$ {}^{1}\eta_1^1 = \{\overline{B,C};\ \overline{D,E,F}\} \qquad {}^{1}\eta_4^1 = \{\overline{A,B,C};\ \overline{D,E,F}\} \qquad {}^{2}\eta_3^1 = \{\overline{A,B};\ \overline{D,E,F}\} $$

$$ {}^{1}\eta_2^1 = \{\overline{\emptyset};\ \overline{A,B,D,F}\} \qquad {}^{2}\eta_1^1 = \{\overline{D,E};\ \overline{A,B,C}\} \qquad {}^{2}\eta_4^1 = \{\overline{D,F};\ \overline{A,B,C}\}. $$

$$ {}^{1}\eta_3^1 = \{\overline{D,E};\ \overline{A,B}\} \qquad {}^{2}\eta_2^1 = \{\overline{D,F};\ \overline{A,B,C}\} $$

Step 4:

$$ {}^{1}f_2^1 = 0 \qquad {}^{2}f_2^1 = {}^{2}f_4^1 $$

$$ {}^{2}f_1^1 = {}^{1}f_3^1 = \tau_2' $$

Thus ${}^{1}f_1^1$, ${}^{2}f_4^1$, ${}^{2}f_3^1$, and ${}^{1}f_4^1$ must still be found.

Step 5:

$$ {}^{1}P_1^1(BC) = \{\overline{B,C};\ \overline{D,E,F}\} \leq \tau_2 \cdot \tau_3' $$

$$ {}^{2}P_4^1(DF) = \{\overline{D,F};\ \overline{A,B,C}\} \leq \tau_1' \cdot \tau_3 $$

$$ {}^{1}P_4^1(BC) = \{\overline{B,C};\ \overline{D,E,F}\} \leq \tau_2 \cdot \tau_3' $$

$$ {}^{1}P_4^1(AB) = \{\overline{A,B};\ \overline{D,E,F}\} \leq \tau_1 \cdot \tau_2 $$

$$ {}^{2}P_3^1(AB) = \{\overline{A,B};\ \overline{D,E,F}\} \leq \tau_1 \cdot \tau_2 $$

The next state expressions for I_1 are:

$$ S_1 = y_2 y_3' I_1 \qquad R_1 = y_2' I_1 $$

$$ S_2 = 0 \qquad R_2 = y_1' y_3 I_1 $$

$$ S_3 = y_2' I_1 \qquad R_3 = y_1 y_2 I_1 $$

$$ S_4 = (y_2 y_3 + y_1 y_2) I_1 \qquad R_4 = y_1' y_3 I_1 . $$

IV. CONCLUSION

An improved algorithm for generating the design equations for asynchronous sequential circuits has been presented. The method has several advantages over standard synthesis techniques which involve state and transition tables (8), as the formation of these tables is circumvented. The method also has several advantages over previous techniques presented by Maki, et. al. (2) and Tan (3). Among these are a much more general selection of circuit realization elements and once the next state partitions are formed many of the coefficient terms may be determined by inspection. The total number depends on the technique used to select the state assignment. The remaining terms are found by observing the relationship between the partitions which have been defined.

V. REFERENCES

1. R. J. Smith, J. H. Tracey, W. L. Schoeffel, and G. K. Maki, "Automation in the Design of Asynchronous Sequential Circuits", 1968 Spring Joint Computer Conf., AFIPS Proc., Vol. 32, Washington, D.C.; Thompson, 1968, pp. 53-60.
2. G. K. Maki, J. H. Tracey, and R. J. Smith II, "Generation of Design Equations in Asynchronous Sequential Circuits", IEEE Trans. on Computers, Vol. C-18, pp. 467-472, May 1969.
3. C. J. Tan, "State Assignments for Asynchronous Sequential Machines", IEEE Trans. on Computers, Vol. C-20, pp. 382-391, Arpil 1971.
4. G. K. Maki, and J. H. Tracey, "State Assignment Selection in Asynchronous Sequential Circuits", IEEE Trans. on Computers, Vol. C-19, pp. 641-644, July 1970.
5. G. K. Maki, D. H. Sawin III, and B. A. Jeng, "Improved State Assignment Selection Tests", IEEE Trans. on Computers, Vol. C-21, pp. 1443-1449, December 1972.
6. J. H. Tracey, "Internal State Assignments for Asynchronous Sequential Machines", IEEE Trans. on Electronic Computers, Vol. EC-16, pp. 551-560, August 1966.
7. S. H. Unger, Asynchronous Sequential Switching Circuits, Wiley-Interscience, New York, 1969.
8. E. J. McCluskey, Introduction to the Theory of Switching Circuits, McGraw-Hill Inc., New York, 1965.
9. M. P. Marcus, Switching Circuits for Engineers, Prentice-Hall Inc., Englewood Cliffs, N.J., 1962, 1967.

A DESIGN TECHNIQUE FOR TOTALLY-SELF-CHECKING ASYNCHRONOUS CIRCUITS*

S. M. REDDY
Division of Information Engineering
The University of Iowa
Iowa City, Iowa 52242

ABSTRACT

A design technique for totally-self-checking asynchronous circuits is given. The design technique given is applicable to normal mode flow tables.

SUMMARY

In this paper we give a technique to design totally-self-checking asynchronous circuits for unidirectional faults. We assume that the flow table to be realized is a normal mode flow table. We give the proposed design technique for Moore type flow tables (the technique is extendable in a straightforward manner to Mealy type flow tables).

The block diagram of the proposed asynchronous circuits is shown in Figure 1. The following assumptions are made on the inputs, state assignments, etc.

(a) The inputs are two-rail; i.e., an input and its complement are independently available. When an input changes it is assumed that the input and its complement change simultaneously.

(b) The state assignment is the single transition time assignment [2] given for fail-safe asynchronous circuits by Metze and Patterson [1]. Specifically we are assuming that the stable states are binary 2K-tuples which are obtained by concatenating a binary K-tuple and its complement.

(c) The outputs are encoded in an appropriate m-out-of-n code.

Definition: The asynchronous network given in Figure 1 is said to be totally-self-checking in the presence of unidirectional faults if the output sequences for all input sequences are either correct or else a non-code output at the output of one of the checkers occurs and furthermore the presence of these faults is detectable (by a non-code output at the output of a checker) with the test inputs possible under either normal operation of the asynchronous circuit or by any extra inputs made available to facilitate testing.

The proposed technique is illustrated for the example given below.

Inputs x_1x_2 / Present State	00	01	11	10	output z
a	ⓐ	d	c	ⓐ	0
b	a	ⓑ	ⓑ	d	0
c	ⓒ	b	ⓒ	a	1
d	c	ⓓ	b	ⓓ	1

---

*The research reported has been supported by NSF Grant ENG75-08570

The state assignment is given below [1].

	y_1	y_2	y_3	$y_1^\dagger$	$y_2^\dagger$	$y_3^\dagger$
a	1	1	1	0	0	0
b	1	0	0	0	1	1
c	0	0	1	1	1	0
d	0	1	0	1	0	1

The design equations for Y_i's can be written by several techniques such that Y_i's are positive unate function of y_i's and inputs (since we are assuming that double rail inputs are available) such that the combinational networks can be constructed from AND and OR gates only [3-5]. One such set of equations is given below.

$$Y_1 = y_1 \bar{x}_1 \bar{x}_2 + y_2^\dagger \bar{x}_1 x_2 + y_3^\dagger x_1 x_2 + y_3 x_1 \bar{x}_2$$

$$Y_2 = y_1 \bar{x}_1 \bar{x}_2 + y_2 \bar{x}_1 x_2 + x_1 \bar{x}_2$$

$$Y_3 = \bar{x}_1 \bar{x}_2 + y_3 x_1 x_2 + y_3 x_1 \bar{x}_2$$

$$Y_1^\dagger = y_1^\dagger \bar{x}_1 \bar{x}_2 + y_2 \bar{x}_1 x_2 + y_3 x_1 x_2 + y_3^\dagger x_1 \bar{x}_2$$

$$Y_2^\dagger = y_1^\dagger \bar{x}_1 \bar{x}_2 + y_2^\dagger \bar{x}_1 x_2 + x_1 x_2$$

$$Y_3^\dagger = \bar{x}_1 x_2 + y_3^\dagger x_1 x_2 + y_3^\dagger x_1 \bar{x}_2$$

$$Z = y_1^\dagger$$

$$\bar{Z} = y_1$$

The totally-self-checking circuit for the design equations given above is given in Figure 2. The checker I of Figure 2 is shown in detail in Figure 3. The extra two inputs S_1 and S_2 are held at $S_1 = 0$ and $S_2 = 1$ during the normal operation of the circuit. S_1 and S_2 are used dynamically only while testing the checker. By arguments similar to the ones used earlier [5] we can prove that the network given in Figure 2 is totally-self-checking except for the self-checking property of checker I (given in detail in Figure 3). The self-checking property of checker I can be established by showing that the following eight tests detect all unidirectional faults in the checker

$$T = \left\{ \begin{array}{cccccccc} S_1 & S_2 & y_1 & y_2 & y_3 & y_1' & y_2' & y_3' \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{array} \right\}$$

The proof that T is indeed a valid test set can be derived by observing that each cell of Figure 3 is a comparator for binary vectors of length 2.

In the general case the operation of the circuits of Figure 1 follows the procedure given below. Notice that the gates in the combinational network and the two checkers are only AND and OR gates.

Initial Testing Phase:

(i) Tests for faults in the asynchronous circuit of Figure 1, except the checkers, are derived by normal procedures [7]. If any faults are not detectable the corresponding leads are removed. In deriving these tests one can assume that all the state variables are being observed.

(ii) Tests for checker II are designed depending on its realization [8,9].

(iii) Checker I is shown in detail in Figure 4. A sufficient test set for this checker to detect all unidirectional faults can be shown to be the following:

A set of states such that each y_i is exercised through a zero and 1 and with the circuit in each one of these states S_0 and S_1 inputs are changed from 0,1 to 1,0 respectively.

(iv) The complete test set for the asynchronous circuit comprises of the input sequences required to take the asynchronous network through a sequence of states such that the tests in (i), (ii) and (iii) are applied. While testing the circuit the z outputs and the outputs of the checkers are observed.

Operation of the Circuit:

(i) Testing mode. The test set derived in (iv) above is applied.

(ii) Normal operation. Under normal operating condition S_0 and S_1 are held at 0,1 or 1,0 respectively.

The results given in this paper can be extended to asynchronous circuits realized by other procedures, for example the circuits realized by procedure given in 3. Also if only single faults are assumed and it is further assumed that the primary inputs are fault-free then double-rail inputs are not necessary.

REFERENCES

[1] W. W. Patterson and G. A. Metze, "A Fail-Safe Asynchronous Sequential Machine," IEEE Trans. Computers, April 1974.

[2] J. H. Tracey, "Internal State Assignments for Asynchronous Switching Circuits," IEEE Trans. Comp., April 1966.

[3] D. H. Sawin and G. K. Maki, "Asynchronous Sequential Machines Designed for Fault Detection," IEEE Trans. on Computers, March 1974.

[4] G. K. Maki, et al., "Improved State Assignment Selection Tests," IEEE Trans. on Computers, December 1972.

[5] M. Diaz, "Design of Totally Self Checking and Fail Safe Sequential Machines," Digest of FTC-4, June 1974.

[6] D. A. Anderson, "Design of Self-Checking Digital Networks Using Coding Techniques," Report R-527, Coordinated Science Laboratory, September 1971.

[7] A. D. Friedman and P. R. Menon, "Fault Detection in Digital Circuits," Prentice-Hall,

[8] D. A. Anderson and G. Metze, "Design of Totally Self-checking Check Circuits for m-out-of-n Codes," IEEE Trans. on Comp., Vol. C-22, pp. 263-269, March 1973.

[9] S. M. Reddy, "Some Results on Totally Self-Checking Checkers," in Proc. 10th Ann. Allerton Conf. Circuit and System Theory, October 4-6, 1972.

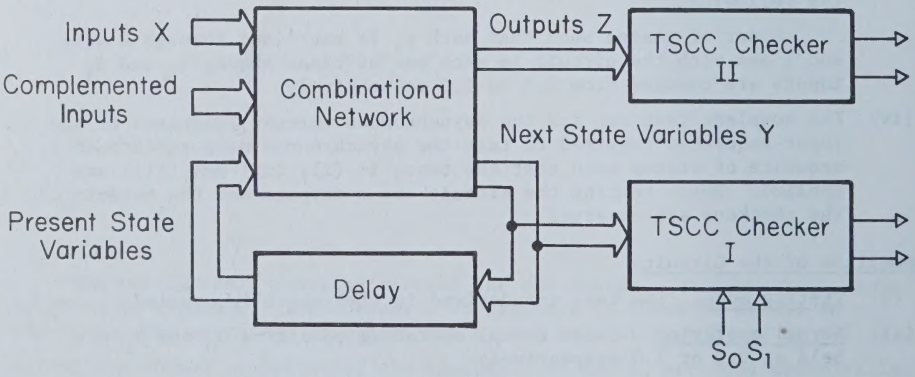


Figure 1. Block diagram of TSC asynchronous circuits.

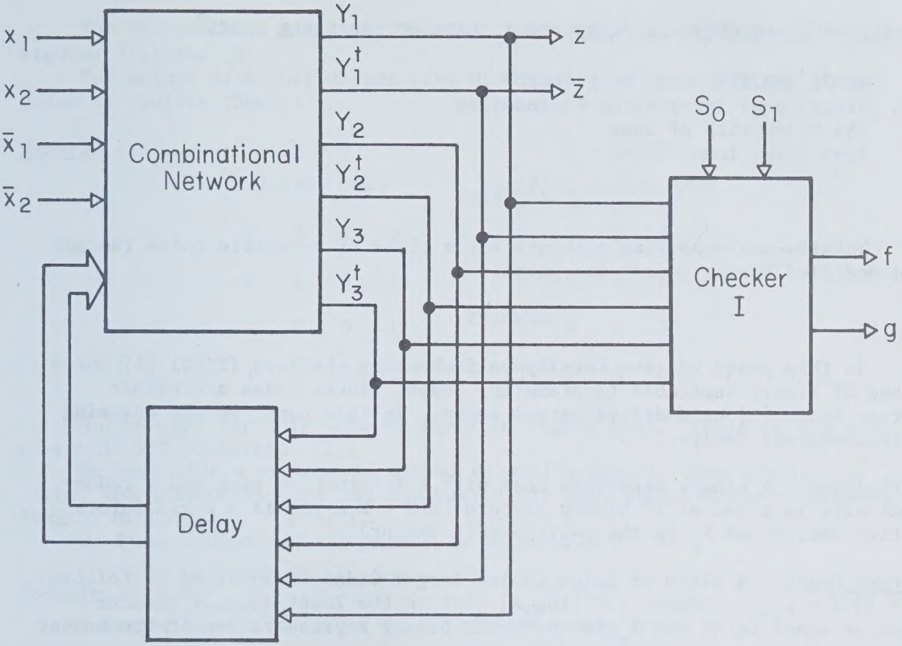

x_1
x_2
$\bar{x}_1$
$\bar{x}_2$
Combinational Network
Y_1
$Y_1^\dagger$
Y_2
$Y_2^\dagger$
Y_3
$Y_3^\dagger$
z
$\bar{z}$
S_0
S_1
Checker I
f
g
Delay

Figure 2. TSC realization for the flow table of Example I.

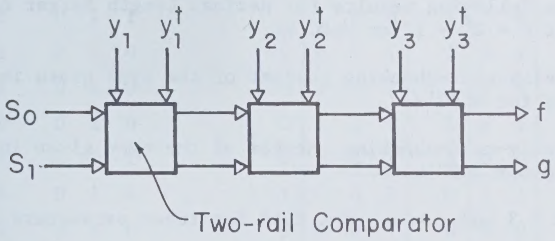

y_1
$y_1^\dagger$
y_2
$y_2^\dagger$
y_3
$y_3^\dagger$
S_0
S_1
f
g
Two-rail Comparator

Figure 3. Checker I for the Example.

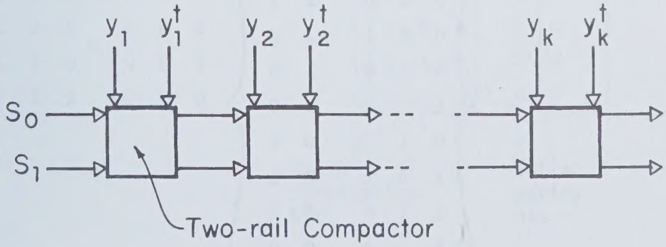

y_1
$y_1^\dagger$
y_2
$y_2^\dagger$
y_k
$y_k^\dagger$
S_0
S_1
Two-rail Compactor

Figure 4. Checker I for Figure 1.

TOTALLY-SELF-CHECKING CHECKERS FOR A CLASS OF SEPARABLE CODES*

M. J. ASHJAEE and S. M. REDDY
Division of Information Engineering
The University of Iowa
Iowa City, Iowa 52242

ABSTRACT

Totally-self-checking checkers for a class of separable codes (Berger and Modified Berger Codes) are given.

SUMMARY

In this paper we give totally-self-checking checkers (TSCC) [1] for a class of binary separable (systematic) codes. These codes are either Berger codes [2] or modified Berger codes. In this paper we are assuming unidirectional faults.

Definition: A binary separable code with k information bits and r redundant bits is a set of 2^k binary vectors $\{X | X = X_I X_r, X_I$ is a k-bit information vector and X_r is the r-bit parity vector$\}$.

Berger Codes: A class of codes called Berger Codes is obtained as follows. $C = \{X : X = X_I X_r\}$ where $r = \lceil \log_2 k \rceil$ ($\lceil x \rceil$ is the least integer greater than or equal to X) and X_r is the r-bit binary representation of the number of zeros in X_I.

Definition: A Berger code C is a Maximal Length Berger code if k has the maximum value for given r.

We derive the following results for Maximal Length Berger codes (MLBC). Note that $K = 2^r - 1$ for MLBC's.

Theorem 1: A totally-self-checking checker of the type given in Figure 1 can be constructed for MLBC's.

Theorem 2: A totally-self-checking checker of the type given in Figure 2 can be constructed for MLBC's.

Example 1: Let k = 3 and r = 2. The MLBC for these parameters is given below:

$$C = \left\{\begin{array}{ccccc} \multicolumn{3}{c}{\leftarrow k \rightarrow} & \multicolumn{2}{c}{\leftarrow r \rightarrow} \\ A & B & C & D & E \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \end{array}\right\}$$

---

*The research reported has been supported by NSF Grant ENG75-08570

The two checkers given in Theorems 1 and 2 for this code are given in Figures 3(a) and (b).

For values of k that do not give MLBC's we need to modify the Berger codes to realize TSCC's.

Example 2: $k = 2$ $r = 2$

Berger Code				Modified Berger Code			
				A	B	C	D
0	0	1	0	0	0	1	1
0	1	0	1	0	1	1	0
1	0	0	1	1	0	0	1
1	1	0	0	1	1	0	0

The checker for this code is given in Figure 4(a). (This is the same as a two-rail comparator [3]).

We next give a systematic method to modify Berger Codes with $k = 2^{r-1}$ (i.e., one information bit and one parity bit more than the closest (in length) MLBC).

We first illustrate the method by an example.

Example 3: $k = 4$ $r = 3$

Berger Code							Modified Berger Code						
0	0	0	0	1	0	0	0	0	0	0	1	1	1
0	0	0	1	0	1	1	0	0	0	1	1	1	0
0	0	1	0	0	1	1	0	0	1	0	1	1	0
0	1	0	0	0	1	1	0	1	0	0	1	1	0
0	0	1	1	0	1	0	0	0	1	1	1	0	1
0	1	0	1	0	1	0	0	1	0	1	1	0	1
0	1	1	0	0	1	0	0	1	1	0	1	0	1
0	1	1	1	0	0	1	0	1	1	1	1	0	0
1	0	0	0	0	1	1	1	0	0	0	0	1	1
1	0	0	1	0	1	0	1	0	0	1	0	1	0
1	0	1	0	0	1	0	1	0	1	0	0	1	0
1	1	0	0	0	1	0	1	1	0	0	0	1	0
1	0	1	1	0	0	1	1	0	1	1	0	0	1
1	1	0	1	0	0	1	1	1	0	1	0	0	1
1	1	1	0	0	0	1	1	1	1	0	0	0	1
1	1	1	1	0	0	0	1	1	1	1	0	0	0

the MLBC with $k = 3$ and $r = 2$ repeated.

Extra information bit

Extra parity bit

The following three points are to be noted.

(i) The parity bits of the information vectors with a zero in the left-most position are changed by adding 011 (binary representation of $3 = 2^{r-1} - 1 = 2^{3-1} - 1$).

(ii) The parity bits of the information vectors with a one in the left-most position are not modified.

(iii) The modified code can be considered to be obtained by taking two copies of the MLBC with k = 3 and r = 2 and annexing an extra information bit and an extra parity bit (whose value is the complement of the value of the information bit).

The TSCC for the code of Example 3 is given in Figure 4(b).

In general the code words of the $k = 2^{r-1}$ Berger codes are modified by adding $2^{r-1} - 1$ to the parity vector of the information vectors with a zero in the left-most position. The modified codes can be readily shown to be equivalent to taking two copies of a $k = 2^{r-1} - 1$ MLBC and annexing an information and a parity bit as indicated in (iii) above.

We can prove the following theorem for these codes.

Theorem 3: For the $k = 2^{r-1}$ Berger codes modified as indicated above a TSCC of the type given in Figure 1 can be constructed.

REFERENCES

1. D. A. Anderson and G. Metze, "Design of Totally Self-checking Check Circuits for m-out-of-n Codes," IEEE Trans. on Computers, March 1973.

2. J. M. Berger, "A Note on Error Detection Codes for Asymmetric Channels," Information and Control, March 1961.

3. D. A. Anderson, "Design of Self-Checking Digital Networks Using Coding Techniques," Report R-527, Coordinated Science Laboratory, September 1971.

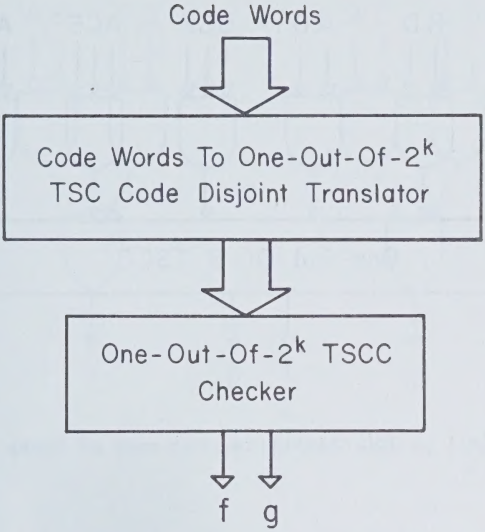

Code Words
Code Words To One-Out-Of-2^k TSC Code Disjoint Translator
One-Out-Of-2^k TSCC Checker
f
g

Figure 1: A TSC Checker for MLBC's.

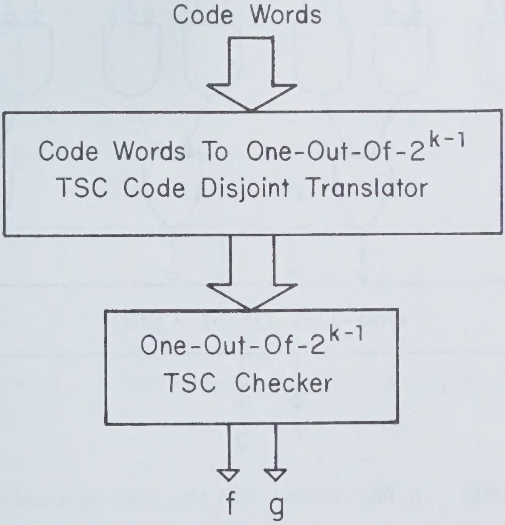

Code Words
Code Words To One-Out-Of-2^{k-1} TSC Code Disjoint Translator
One-Out-Of-2^{k-1} TSC Checker
f
g

Figure 2: A TSC Checker for MLBC's.

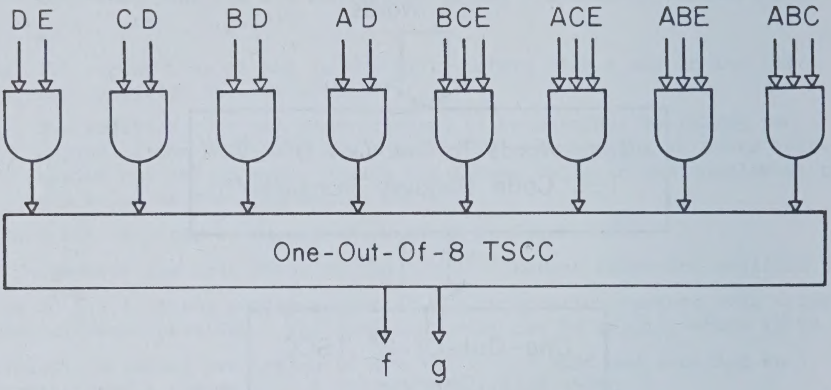

D E
C D
B D
A D
BCE
ACE
ABE
ABC
One-Out-Of-8 TSCC
f
g

Figure 3(a): A TSC Checker for the code of Example 1.

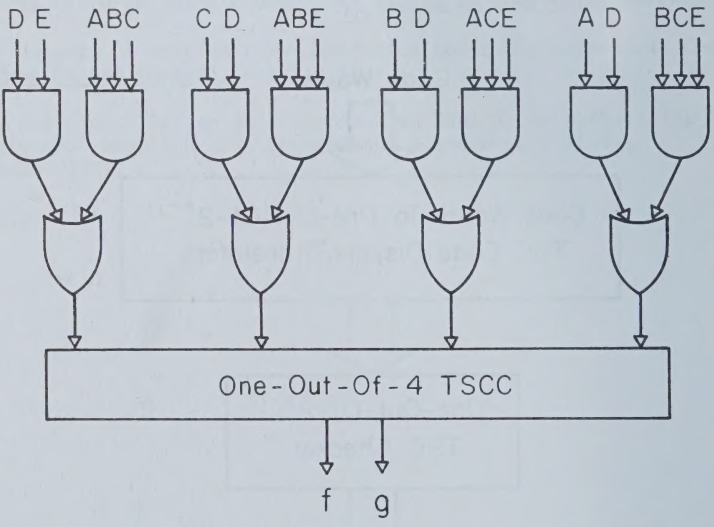

D E
ABC
C D
ABE
B D
ACE
A D
BCE
One-Out-Of-4 TSCC
f
g

Figure 3(b): A TSC Checker for the code of Example 1.

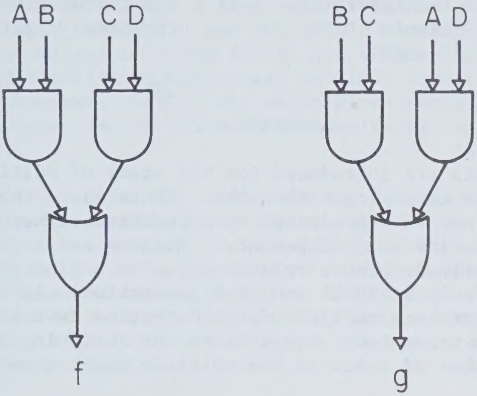

A B
C D
B C
A D
f
g

Figure 4(a): A TSC Checker for the code of Example 2.

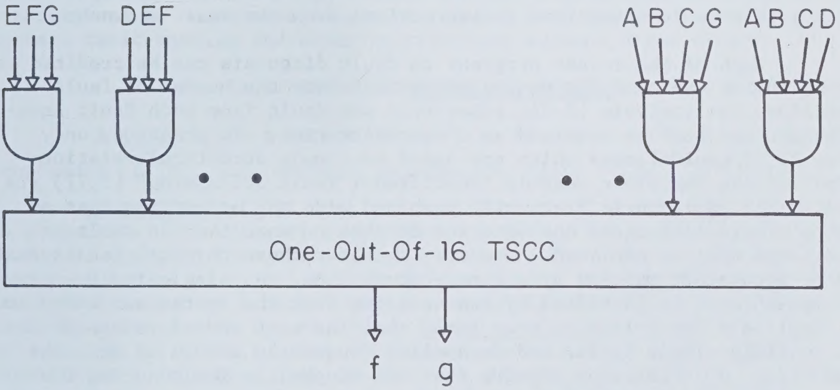

E F G
D E F
A B C G
A B C D
One-Out-Of-16 TSCC
f
g

Figure 4(b): A TSCC Checker for the code of Example 3.

MULTIPLE FAULT DIAGNOSIS IN COMBINATIONAL NETWORKS

GERNOT METZE
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois 61801

CHARLES CHA
IBM Corporation
Yorktown Heights, New York

ABSTRACT

Prime faults are introduced for the study of multiple fault diagnosis in combinational logic networks. It is shown that every multiple fault in a network can be represented by a functionally equivalent fault with prime faults as its only components. Masking and covering relations among faults are defined. These relations can be exploited to greatly simplify multiple fault analysis and test generation. An efficient algorithm that generates a multiple fault detection test set and identifies any redundancies is presented. Suggestions for designing networks to yield a minimum number of tests in the multiple fault detection test set are included.

1. INTRODUCTION

Many physical failures in combinational networks can be modeled logically as a set of lines in the network that are permanently stuck-at-0 or stuck-at-1 [1,2]. This fault model can be viewed as a transformation that preserves the logical nature of the network and thus facilitates analysis and simulation of the faulty networks. Since there are 3^n-1 modeled fault conditions in a network with n lines, a direct analysis of all possible fault conditions is impractical even for small networks.

Much of the recent progress in fault diagnosis can be credited to the extensive use of fault equivalences to reduce the number of fault conditions for analysis [3-7], since only one fault from each fault equivalence class need be retained as a representative. In practice, only those fault equivalences which are based on simple structural relationships and are therefore readily identified ("fault collapsing" [3,7]) are used. This approach is frequently combined with the assumption that at most a single-line fault has occurred so that no more than 2n fault conditions need be analyzed. However, even for those physical faults that can be accurately modeled by a single stuck line, the single-fault assumption must be justified by guaranteeing that the system was known to be completely fault-free at some time, that the test procedure checks for all possible single faults and is applied frequently enough to make the probability of occurrence of more than one stuck-line fault during the interval between tests, which the test procedure might miss, sufficiently small. For arbitrary networks, the single-fault assumption is therefore invalid for initial acceptance testing, or for routine testing for faults that cannot be modeled by single stuck lines, e.g. certain shorts, bridging faults, or secondary induced faults. In those situations a consideration of multiple faults, i.e. faults with several stuck-line fault components, cannot be avoided.

---

*This work was supported by the Joint Services Electronics Program (U. S. Army, U. S. Navy, and U. S. Air Force) under Contract DAAB-07-72-C-0259.

Several approaches to the diagnosis of multiple faults have been presented in the literature. Test sets derived under the single-fault assumption generally are capable of detecting a large number of multiple faults, but attempts to convert a single-fault detection test set (SFDTS) into a multiple-fault detection test set (MFDTS) by augmenting the SFDTS with tests for the missed multiple faults are frequently impractical due to the difficulty of identifying the missed multiple faults [1,2,4,8]. It should be noted, however, that if the network has certain special structural properties, then detection of all multiple faults by an SFDTS can be guaranteed [3-5].

For arbitrary networks, a significant reduction in the number of fault conditions under consideration can be achieved by using "third stage fault collapsing" techniques [3] to identify the "check points" [6] of the network such that any fault in the network is representable by an equivalent fault whose components are checkpoint faults only. Since the main goal is an analysis of multiple-line faults, single-line faults are treated as degenerate cases of multiple-line faults and may in fact be represented by multiple-checkpoint faults.

We claim that multiple-fault diagnosis can be simplified considerably further by using "prime faults" [10] instead of checkpoint faults to represent any multiple-line fault in the network. Although there are in general somewhat more prime fault sites than checkpoints in a network, each prime fault site can be in only 2 states whereas each checkpoint has 3 states. Thus, the Boolean manipulation of prime fault functions is much easier than that of checkpoint fault functions and, the number of possible multiple-line fault conditions is a priori lower for the prime fault approach than for the checkpoint approach. The number of prime fault combinations to be considered can be reduced even further by exploiting certain fault masking and covering relations between prime faults.

2. FAULT REPRESENTATIONS AND RELATIONS

A *fault* F_i in a network N consists of one or more stuck-line components $\{f_1^{\alpha}, f_1^{\beta}, \ldots\}$, where f_1^{α} denotes the fault "line 1 stuck-at-α." F_i is a *proper fault* if it does not contain both f_n^0 and f_n^1 for any line n as components; otherwise F_i is undefined. Hence, two proper faults F_i and F_j may be *combined* into the fault F_iF_j only if the union of the sets of component faults of F_i and F_j is a proper fault. Let $Z[X,\emptyset]$ denote the fault-free output of N under input X and $Z[X,F_i]$ the output of N under fault F_i. Two faults F_i and F_j are *functionally equivalent*, $F_i \equiv F_j$, if and only if $Z[X,F_i] = Z[X,F_j]$ for all input vectors X. An input vector t constitutes a *detection test* for fault F_i if and only if $Z[t,F_i] \neq Z[t,\emptyset]$. A fault F_j is *undetectable* if and only if $Z[X,F_j] = Z[X,\emptyset]$ for all input vectors X. An undetectable fault with r stuck-line components is equivalent to an *r-line redundancy* in the network. R-line redundant networks are not necessarily redundant for fewer than r lines.

Multiple fault diagnosis ultimately involves the analysis of all single-line faults and their combinations or their equivalents. The approach presented here is based on the use of a particular subset of the single-line faults of the network, the *prime faults*, selected such that any fault in the network can be represented by an equivalent multiple-line fault consisting only of prime faults as components. The prime faults of a network are found by Procedure 1:

Procedure 1: Selection of the Set of Prime Faults for a Combinational Network

1) Place a prime fault on every gate input line that is a fanout branch line or a primary input line that does not fan out, skipping inverters (unless the output of the inverter is a primary output, then treat it as a single-input NAND or NOR gate). The prime fault is the stuck-at-1 fault for AND or NAND gate inputs, the stuck-at-0 fault for OR or NOR gate inputs ("dominant" input faults).

2) Identify as prime gate every gate that has faults placed on all its inputs by step 1). Place a prime fault on the output line of every prime gate that does not fan out: The prime fault is the stuck-at-1 fault for OR or NAND gate outputs, the stuck-at-0 fault for AND or NOR gate outputs.

The selection of prime faults, similar to that of checkpoints [6], is based essentially on "third stage fault collapsing" techniques [3] to systematically replace (i) component faults within fanout-free subnetworks by equivalent faults involving only the subnetwork inputs, i.e. primary inputs or fanout branches, (ii) dominant faults on the inputs to "prime" gates by faults on the outputs of the prime gates, and (iii) component faults on prime gate outputs that fan out by the corresponding faults on the fanout branches. It is to be noted that the number of prime faults in a network is generally less than the number of single faults, since many single faults can be represented by equivalent multiple prime faults.

The set of prime faults chosen by Procedure 1 is not necessarily minimal; in networks where prime AND gates feed AND or NAND gates, or prime OR gates feed OR or NOR gates, a subset of prime faults can be replaced by a single prime fault at a different line. The modification of the prime fault selection rules to account for such special cases has been omitted here for simplicity. The prime fault approach differs from the checkpoint approach [6] in two important aspects:

In the checkpoint approach, both the stuck-at-0 and the stuck-at-1 fault are considered at each checkpoint. Consequently, there are 3^c-1 possible fault conditions involving c checkpoints. Also, the three-state nature of the checkpoint variables makes their Boolean manipulation somewhat cumbersome. In the prime fault approach, only one of the stuck-at faults is considered at each prime fault site. Thus, each line is in at most two states. Hence, Boolean manipulation of prime fault variables is straightforward and, incidentally, combinations of prime faults always result in proper faults, because no combination can involve both the stuck-at-0 and the stuck-at-1 fault of one line. Further, there are at most 2^p-1 fault conditions for p prime faults and, although $p > c$ generally, it is easy to show that $p \leq 3c/2$ and therefore $2^p-1 < 3^c-1$ for $c > 1$. Consider e.g. a network consisting of a k-input AND gate: it has k checkpoints vs. (k+1) prime faults, hence $p \leq (k+1)c/k$ where k is the minimum number of inputs to any prime gate in the network.

Perhaps the most important advantage of prime faults is that certain simple relations between faults can be exploited to drastically reduce the number of fault combinations for analysis. The discussion that follows applies to any single fault or fault component and in particular to prime faults.

A fault F_i m-covers a fault F_j if $Z[X,F_i] = Z[X,F_iF_j]$ for all input vectors X, i.e. if $F_i \equiv F_iF_j$. If a given fault F_i is decomposed into two subfaults F_k and F_j such that $F_i = F_kF_j$, F_k m-covers F_j, and moreover F_k is the smallest subset of F having these properties, then F_k is called the kernel or the canonical fault representation of F_i. Since $Z[X,F_k] = Z[X,F_i]$ for all X, the kernel F_k can always be used to represent F_i and the subfault $F_j = F_i - F_k$ may be ignored.

It can be shown easily [10] that if F_i m-covers and is m-covered by F_j, then $F_i \equiv F_j$. However, the converse does not hold, and m-covering neither implies nor is implied by fault dominance [3,7]. It should also be noted that functional equivalence is not a congruence relation, i.e. $F_1 \equiv F_2$ and $F_3 \equiv F_4$ does not imply $F_1F_3 \equiv F_2F_4$. Therefore, prime faults that are equivalent may not be eliminated from the prime fault set.

A test t_i designed to detect fault F_0 in general also detects several other faults; conversely it is possible that the occurrence of other faults interferes with the detection of F_0 by test t_i. A fault F_1 masks a fault F_0 under test t_i if F_0 but not F_1F_0 is detected by test t_i, i.e. if $Z[t_i,F_0] \neq Z[t_i,\emptyset] = Z[t_i,F_1F_0]$. Masking conditions can be expressed as a fault detection expression Δ_i associated with test t_i; for this example $\Delta_i = F_0 \cdot \overline{F}_1$. This expression can be found directly from the network during the test generation process, or as $\Delta_i = Z[t_i,\emptyset] \oplus Z[t_i,F]$ [9].

The fault detection function of a test set is formed as the Boolean sum of the detection expressions for each test in the test set, reduced by standard simplification methods. A test set is an MFDTS if it detects all single faults and all possible combinations of single faults, i.e. if the detection function reduces to $\Sigma_{(i)}\Delta_i = F_1 + F_2 + F_3 + \cdots F_n = \Sigma_{(j)}F_j$ for n possible single faults. If the reduced detection function contains fault masking cycles, e.g. a cycle of the form $F_1\overline{F}_2 + F_2\overline{F}_3 + \cdots + F_q\overline{F}_1$, then the multiple fault $F_1F_2F_3 \ldots F_q$ will not be detected and the test set is not an MFDTS.

Attempts to break each fault masking cycle may involve an exhaustive search for alternate tests with different masking conditions. If a masking cycle cannot be broken, the undetectable multiple fault $F_1F_2F_3 \ldots F_q$ of the masking cycle corresponds to a multiple-line redundancy in the network.

Obviously, the test generation procedure should derive individual tests to detect faults with the fewest masking conditions to enhance the potential multiple-fault detection capability of the test set.

Not all fault masking conditions need in fact be examined. If F_1 masks F_0 under t_i but F_1 also m-covers F_0, then F_0 cannot mask F_1 under any test designed to detect F_1. Therefore, if F_1 is detected, then the combination fault F_0F_1 cannot escape detection. As a consequence, only c-masking faults, i.e. faults that mask but do not m-cover another fault, have to be retained in the fault detection expression. Masking, m-covering, and c-masking relations can be identified from the structure of the network at the time of test generation. Using prime faults rather than arbitrary single-line faults and exploiting m-covering relations to eliminate many masking conditions from consideration thus can greatly simplify the task of analyzing the multiple-fault detection capability of a test.

3. AN ALGORITHM FOR GENERATING A MULTIPLE FAULT DETECTION TEST SET

An MFDTS can be constructed by first generating a test set to detect the prime faults one at a time, identifying the combinations of prime faults that are not detected, and generating tests for them. This procedure does not differ from the classical approach in concept but differs in execution, since there are fewer prime faults than single faults and since masking and covering relations between prime faults can be used to reduce the number of combinations of prime faults to manageable size.

A test detects a fault by setting up one or more sensitized paths from the fault site to an observable output. Each gate on the sensitized path must have the off-path inputs at the propagating value P ($P=1$ for ANDs and NANDs, $P=0$ for ORs and NORs) so that the input on the sensitized path controls the gate output. The on-path inputs are either at the propagating value P or the blocking value B ($B=0$ for ANDs and NANDs, $B=1$ for ORs and NORs). The corresponding gate input vectors, e.g. PP...P and BP...P, will be called P-vectors and B-vectors. In networks with reconvergent fanout several reconvergent paths may be sensitized simultaneously; the on-paths inputs at the reconvergence gate must then have identical values to control the gate output. Such gate input vectors will be called P^*- and B^*-vectors. To avoid conflicts, sensitization of multiple paths in reconvergent fanout networks should be allowed to occur naturally. That is, when the selected path is sensitized through a branch of a reconvergent fanout, other branches should neither be deliberately sensitized nor deliberately blocked; inputs on the other branches are not assigned until after all conditions for sensitization of the selected path are fulfilled.

When the tested fault occurs the role of P- and B-vectors is reversed along the path, thus propagating the fault indication to the observable output. However, the occurrence of other faults may interfere with this process. Let gates be identified as P-gates, etc., by the input vector they receive when the tested fault is not present. Now the output of a P- or P^*-gate changes when the fault causes the on-path inputs to change from P to B. Clearly, the change of additional, off-path input values from P to B cannot mask the output change. However, when the on-path inputs to a B- or B^*-gate change from B to P, a simultaneous off-path input change from P to B will mask the output change. Thus, only faults in the subnetworks feeding the off-path inputs to B- or B^*-gates can cause fault-masking. To minimize the number of possible masking faults, "optimally desensitizing" input vectors (all B- or $\overline{P}$-vectors) should be applied to gates of these subnetworks whenever possible. On the other hand, faults in subnetworks feeding off-path inputs to P- or P^*-gates can never cause fault-masking and, in fact, can be detected without difficulty at the same time. Thus, to maximize the number of faults detected by a given test, B-vectors should be applied to gates of these subnetworks whenever possible. An outline of a procedure to generate a test for a given prime fault follows; details may be found in [10]:

Procedure 2: Generation of a Test for a Given Prime Fault

1) Select a path that can be sensitized from the fault site to an observable output. Assign P- and B-vectors along the path to obtain sensitization. Leave off-path branches of reconvergent fanouts unassigned until after all other input assignments have been made. (If no consistent assignment can be found the fault is undetectable. Exit.)

2) Identify all prime faults on the sensitized path. Determine which additional prime faults may be detected along the path and which prime faults cause masking.

3) Identify the subnetworks feeding off-path inputs: Whenever possible, assign $\overline{P}$-vectors to gates in subnetworks feeding B- or B^*-gates, and B-vectors to gates in subnetworks feeding P- or P^*-gates. List the masking conditions contributed by the subnetworks.

4) Find the masking conditions for all other prime faults detected by the test.

Procedure 2 is used repeatedly to form a test set TS that detects all prime faults individually. These tests are locally optimum, in the sense that, for the chosen sensitized path, each test detects the maximum number of prime faults with the minimum number of masking conditions. The test set TS must then be analyzed to identify any combinations of prime faults that are not detected. A procedure for this analysis, Procedure 3, is presented informally by way of an example:

Example 1. For purposes of explanation, it is assumed that a given single-fault detection test set is to be analyzed for its multiple-fault detection capabilities. Consider the network of Figure 1 and the test set T. The prime faults are lines x_1 through x_8 stuck-at-1; let them be denoted by f_1 through f_8. Each test in the test set is now analyzed to determine which prime faults are detected by it and which prime faults cause fault masking. For example, test t_2 sensitizes the path x_2-x_7-x_9-x_{11}; it detects f_2 masked by f_7 or f_4. Thus, the detection expression for test t_2 is $\Delta_2 = f_2(\overline{f_7+f_4})$. However, f_7 also m-covers f_2 and therefore need not be listed among the masking conditions; hence Δ_2 reduces to $\Delta_2 = f_2 \cdot \overline{f}_4$. This expression could also have been derived as follows: Let $Z[X,F]$ denote the output of the network as a function of the input variables x_j as well as of the fault variables f_k. Then the faults detected (and their masking conditions) can be found from the expression $\Delta_i = Z[t_i,\emptyset] \oplus Z[t_i,F]$. $Z[X,F]$ for the network shown is

$$Z[X,F] = (\overline{x}_1\overline{f}_1 + \overline{x}_2\overline{f}_2 + f_7)(x_3 + f_3) + (x_4 + f_4)(\overline{x}_5\overline{f}_5 + \overline{x}_6\overline{f}_6 + f_8).$$

For $t_2 = (101000)$, $Z[t_2,\emptyset] = 1$; thus $\Delta_2 = f_2\overline{f}_7(\overline{f}_4 + f_5f_6\overline{f}_8)$. Because of m-covering, the factor $\overline{f}_7$ can be deleted, resulting in $\Delta_2 = f_2 \cdot \overline{f}_4 + f_2f_5f_6 \cdot \overline{f}_8$. The last term indicates a multiple fault detected by t_2. Such terms may be ignored for the moment since it is hoped that the test set will turn out to be an MFDTS. If the test set does not detect all possible combinations of faults, such terms will have to be examined because they may be needed to break masking cycles.

The fault detection expressions for the six tests, derived from the $Z[X,F]$ function, are

$$\Delta_1 = f_1(\overline{f_7 + f_4f_8})$$

$$\Delta_2 = f_2(\overline{f_7 + f_4}) + f_2f_5f_6(\overline{f_7 + f_8})$$

$$\Delta_3 = f_6(\overline{f_8 + f_3f_7})$$

$$\Delta_4 = f_5(\overline{f_8 + f_3}) + f_1 f_2 f_5(\overline{f_7 + f_8})$$

$$\Delta_5 = f_7 + f_8$$

$$\Delta_6 = f_3(\overline{f}_2) + f_3 f_7 + f_4(\overline{f}_5) + f_4 f_8.$$

The Boolean sum of these expressions is the fault detection function Δ for the test set. Much further manipulation can be saved, however, by immediately eliminating from consideration those masking faults that also m-cover. The following m-covering relations can be determined by inspection of the network:

f_7 m-covers f_1 or f_2; f_8 m-covers f_5 or f_6;

$f_3 f_7$ m-covers f_4, f_5, f_6, f_8; and $f_4 f_8$ m-covers f_1, f_2, f_3, f_7.

Thus, $f_1 \cdot \overline{f}_7$ may be replaced by f_1, $f_1(\overline{f_4 f_8})$ by f_1, etc. In this example, the same reductions can also be justified on the basis that f_7 and f_8 are detected unconditionally. Therefore, the masking conditions can be removed from f_1, f_2, f_3, and f_6 and Δ reduces to

$$\Delta = f_1 + f_2\overline{f}_4 + f_6 + f_5\overline{f}_3 + f_7 + f_8 + f_3\overline{f}_2 + f_4\overline{f}_5.$$

The existence of the f_2-f_4-f_5-f_3 cycle indicates that the test set T is not an MFDTS because it cannot detect the 4-tuple fault $f_2 f_3 f_4 f_5$.

If test t_2 is replaced by $t_2' = (101011)$ with $\Delta_2' = f_2(\overline{f_7 + f_4 f_8})$, which reduces to $\Delta_2' = f_2$ when m-covering masking faults are deleted, then the fault masking cycle is broken and therefore the test set T' is an MFDTS. (It is not surprising that t_2 is a worse test than t_2'; as shown in Figure 1, t_2 applies a B-vector to G_4 which feeds the off-path input to B-gate G_5, whereas t_2' applies the $\overline{P}$-vector to reduce fault masking effects as suggested at the beginning of this section.)

An alternate implementation of Procedure 3 involves a tabular approach which works particularly well when the fault detection and masking information is determined directly from the network, and if at least one fault is detected unconditionally.

Procedure 3': A Tabular Method to Determine the Multiple-Fault Detection Capability of a Test Set

List every prime fault as a column heading and every test as the heading of a set of rows, one row for each prime fault detected. For a test t_i which detects prime fault f_j masked by f_k and by $f_m \cdot f_n$, i.e. for $\Delta_i = f_j(\overline{f_k + f_m f_n})$, enter a 0 in the (i,j) position to indicate that test t_i detects f_j. Different nonzero integers are used to indicate the different masking conditions: Thus, put a 1 in the (i,k) position to indicate that fault f_k masks fault f_j under test t_i, and put 2's in the (i,m) and (i,n) positions to indicate that the fault $f_m \cdot f_n$ also masks fault f_j under test t_i. If another prime fault f_p is also detected by test t_i under exactly the same masking conditions, enter another 0 in position (i,p); otherwise use another row for f_p and test t_i. After all the information about the tests in the test set is recorded, reduce the fault detection table as follows: Delete a nonzero entry in position (i,k), and any other entry of the same value in row i, if there is a 0 in position i',k) and row i' has no nonzero entries remaining, i.e. if test $t_{i'}$ detects fault f_k without masking conditions. Repeat this process until no further entries can be deleted. If there is at least one 0 in every column and no nonzero entries remain in the table, the set of tests is an MFDTS. If some nonzero entries remain in the table, at least one cycle of fault masking conditions exists such that if all the faults in the cycle occur simultaneously they will not be detected by the set of tests under examination.

The tabular analysis of Test Sets T and T' is shown in Tables 1a and 1b.

The complete algorithm for generating a multiple-fault detection test set can now be stated:

Algorithm for Generating an MFDTS for a Combinational Network

1) Identify the prime faults of the network (Procedure 1).

2) Generate a test for each prime fault and identify the masking conditions (Procedure 2). Call the resulting set of tests TS.

3) Examine whether TS also detects all combinations of prime faults (Procedure 3). If it does, TS is an MFDTS and the algorithm is terminated.

4) If TS does not detect all combinations of prime faults, one or more cycles of fault masking conditions exist. Attempt to break each cycle by generating a new test, by Procedure 2, for some fault in the cycle and using the new test in place of, or in addition to, an old test. If a new test can be found return to step 3).

5) If no new tests can be found and an unbroken masking cycle remains, the multiple fault corresponding to the masking cycle is undetectable and the circuit is therefore multiple-line redundant.

Two heuristics can be used to enhance the fault detection capability of a test generated in step 2) of the Algorithm: (1) Undetected faults farthest from the output should be attacked first so that a large number of additional prime faults is detected along the sensitized path; (2) For different faults associated with the same gate, different paths should be sensitized so that new additional prime faults are detected. The application of the algorithm is illustrated by the following example:

Example 2: An MFDTS will be derived for the circuit shown in Figure 2, taken from [4].

1) By Procedure 1, the prime faults are lines x_1 to x_{18} stuck-at-1, denoted by f_1 to f_{18}.

2a) Arbitrarily choose the fault f_1 and the sensitized path x_1-x_{11}-x_{13}-x_{16}-x_{19}-x_{21}. By Procedure 2, using D to represent a value that is 1 normally and 0 if the fault occurs: Set $x_1 = \overline{D}$, $x_2 = x_{12} = x_7 = x_{15} = 1$ and leave x_{20} unassigned since x_{20} is a reconvergent fanout branch. Then $x_{11} = D$, $x_{13} = x_{14} = \overline{D}$, $x_{16} = D$, and $x_{19} = \overline{D}$.

The other inputs at the reconvergence gate are still unassigned; set $x_{20} = 1$.

Since G_3 has a P-input, assign a B-input to G_2: Choose $x_3 = 0$, $x_4 = 1$ so that the prime fault f_3 is also detected by the test. The masking faults for f_3 will be the same as for f_1, since gates at which masking might occur are on the same sensitized path.

Similarly, G_8 has a P-input, thus the B-input $x_5 = 0$, $x_6 = 1$ is assigned to G_4 so that the fault f_5 is also detected by the test. However, in this case the masking faults for f_5 may differ from those for f_1 since the sensitized paths differ.

Since G_{10} has a B-input, $x_{17} = x_{18} = 0$ should be tried first. However, due to existing assignments, $x_{17} = 1$ or D; $x_{17} = D$ is preferred since then $x_{17} = 0$ if f_1 occurs. This requires $x_8 = 1$. Then $x_{18} = 0$ requires $x_9 = x_{10} = 1$. All inputs are now assigned and Test $t_1 = 010\ 011\ 111$ has been generated.

The path $x_{13} - x_{16}$ is a part of a reconvergent fanout branch. Since $x_{13} = \overline{D}$, test t_1 also detects f_{13}. Since $x_{16} = D$ and G_{10} has a B-input but not a B^*-input, f_{16} c-masks f_1 and f_3.

Test t_1 also detects f_5 with the sensitized path x_5-x_{15}-x_{19}. Since G_{10} has a B-input and $x_{17} = 1$ when the fault f_5 occurs, f_{18} c-masks f_5. Similarly, f_{13} with the sensitized path x_{13}-x_{16}-x_{19} is c-masked by f_{18} under t_1.

The fault detection expression for $t_1 = 0101\ 011\ 111$ is

$$\Delta_1 = (f_1 + f_3)\overline{f}_{16} + (f_5 + f_{13})\overline{f}_{18}.$$

2b) The next untested prime fault farthest from the output is f_2. Since f_2 is a prime fault on the same gate as f_1, a new sensitized path, say x_2-x_{11}-x_{14}-x_{17}-x_{20}-x_{21}, is selected and $t_2 = 1010\ 111\ 110$ is generated with $\Delta_2 = (f_2 + f_4)\overline{f}_{17} + (f_{10} + f_{14})\overline{f}_{15}$.

Similarly, the following tests are generated:

for f_4, $t_3 = 1100\ 001\ 011$ which detects $\Delta_3 = f_{11}(\overline{f_{13} + f_5 f_6}) + f_{16} + f_{18}\overline{f}_8$

for f_{12}, $t_4 = 0011\ 110\ 100$ which detects $\Delta_4 = f_{12}(\overline{f_{14} + f_9 f_{10}}) + f_{17} + f_{15}\overline{f}_7$

for f_6, $t_5 = 1111\ 100\ 111$ which detects $\Delta_5 = f_6 + f_7$

and

for f_8, $t_6 = 1111\ 111\ 001$ which detects $\Delta_6 = f_8 + f_9$.

3) The test set $TS = (t_1, t_2, t_3, t_4, t_5, t_6)$ detects all prime faults by construction; whether or not it detects all combinations of prime faults can be determined by Procedure 3:

Using the algebraic method,

$$\Delta = \Delta_1 + \Delta_2 + \Delta_3 + \Delta_4 + \Delta_5 + \Delta_6$$

is formed and simplified by standard Boolean procedures. It is easy to show that Δ reduces to

$$\Delta = f_1 + f_2 + \cdots + f_{17} + f_{18},$$

i.e., TS detects all prime faults in all combinations and is therefore an MFDTS. An application of the tabular method of Procedure 3' to this example is shown in Table 2.

The set of 6 tests derived for this network happens to be optimal. It differs only trivially from the test set derived in [4]; however, the analysis of the multiple-fault detection capability of the derived test set appears to be more straightforward using the prime fault approach presented here. The checkpoint approach [6] would have generated either 13 or 22 tests, depending on whether "abnormal true" or "abnormal false" tests were used.

4. APPLICATIONS, EXTENSIONS, AND CONCLUSIONS

The representation of multiple faults and the algorithm for the generation of a multiple-fault test set proposed here include several important advantages over existing methods: The representation of multiple faults in terms of prime fault components leads to fewer faults and hence fewer fault combinations, permits a straightforward Boolean description of the effect of faults on the network in terms of 2-valued fault variables, and facilitates the identification of masking and m-covering relations between prime faults. The m-covering relation in particular is an important tool for eliminating large numbers of masking fault combinations from further consideration. The test generation algorithm includes heuristics for enhancing the fault-detection capability of each test while reducing the effect of masking faults and, incidentally, a simple method for avoiding common conflicts in reconvergent fanout situations by sensitizing only one path deliberately and permitting the sensitization of other paths to occur naturally.

In addition to the obvious uses of deriving an MFDTS or checking whether a given test set is an MFDTS, the algorithm can be used to identify multiple-line redundancies in the network [10]. The algorithm can be easily adapted to combinational networks having multiple outputs; nonreconvergent fanout branches need not be considered as prime fault sites and tests can be designed to simultaneously sensitize several paths leading to different outputs so that a large number of prime faults is detected at once. The algorithm could also be extended to the generation of multiple-fault location test sets by taking advantage of the fault-class information provided by each test.

Experience with the use of the algorithm suggests that reconvergent fanout tends to increase the number of prime faults, the number of tests in the test set, the difficulties of analyzing the test set for its multiple-fault detection capabilities, and the chances that the network has unintentional redundancies and hence undetectable faults. Thus, the diagnosability of a network can be enhanced by selecting a realization with the minimum number of reconvergent fanouts.

5. REFERENCES

1. H. Y. Chang, E. G. Manning, and G. Metze, Fault Diagnosis of Digital Systems, Wiley-Interscience, 1970.

2. A. D. Friedman and P. R. Menon, Fault Detection in Digital Circuits, Prentice-Hall, 1971.

3. D. R. Schertz and G. Metze, "A New Representation for Faults in Combinational Digital Circuits," IEEE Trans. on Computers, Vol. C-21, pp. 858-866, August 1972.

4. J. P. Hayes, "A Study of Digital Network Structure and Its Relation to Fault Diagnosis," Report R-467, Coordinated Science Laboratory, University of Illinois, Urbana, May 1969.

5. D. R. Schertz and G. Metze, "On the Design of Multiple Fault Diagnosable Networks," IEEE Trans. on Computers, Vol. C-20, pp. 1361-1364, November 1971.

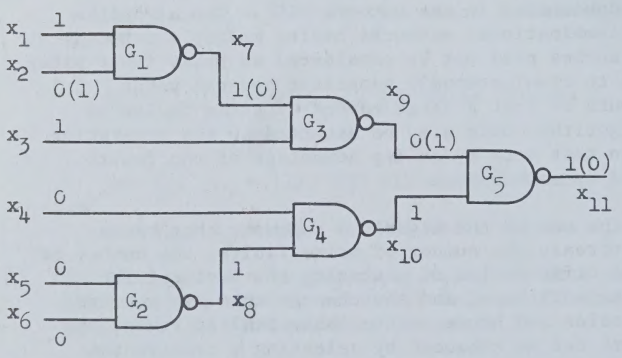
6. D. C. Bossen and S. J. Hong, "Cause-Effect Analysis for Multiple Fault Detection in Combinational Networks," IEEE Trans. on Computers, Vol. C-20, pp. 1252-1257, November 1971.

7. F. W. Clegg and E. J. McCluskey, "Fault Equivalence in Combinational Logic Networks," IEEE Trans. on Computers, Vol. C-20, pp. 1286-1293, November 1971.

8. S. S. Yau and Y. S. Tang, "An Efficient Algorithm for Generating Complete Test Sets for Combinational Logic Circuits," IEEE Trans. on Computers, Vol. C-20, pp. 1245-1251, November 1971.

9. D. R. Schertz, "Applications of a Fault Class Representation to Diagnosis," Paper R73-271, Computer Group Repository, June 1973.

10. C. W-Y. Cha, "Multiple Fault Diagnosis in Combinational Networks," Report R-650, Coordinated Science Laboratory, University of Illinois, Urbana, June 1974.



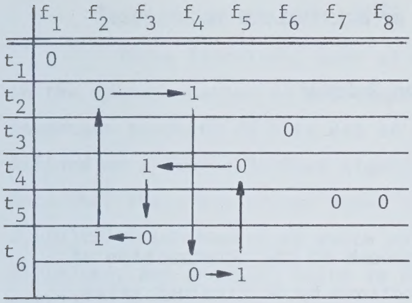
Test Set T

	$x_1x_2x_3x_4x_5x_6$
t_1	0 1 1 0 1 1
t_2	1 0 1 0 0 0
t_3	1 1 0 1 1 0
t_4	0 0 0 1 0 1
t_5	1 1 1 1 1 1
t_6	1 0 0 0 0 1

Test Set T'

	$x_1x_2x_3x_4x_5x_6$
	. . .
t_2'	1 0 1 0 1 1

Figure 1. Network and Test Sets for Example 1.
(Sensitization for Test 2 shown)



	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
t_1	0							
t_2		0 →	→	1				
t_3						0		
t_4			1 ←	←	0			
t_5							0	0
t_6		1 ←	0					
				0 →	1			

Table 1a. Prime Fault Detection Table for Test Set T

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
t_1	0							
t'_2		0						
t_3						0		
t_4			$\cancel{1}$		0			
t_5							0	0
t_6		$\cancel{1}$	0					
				0	$\cancel{1}$			

Table 1b. Prime Fault Detection Table for Test Set T'

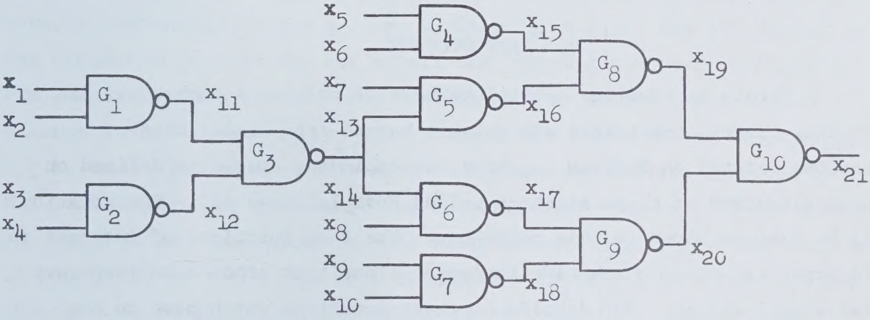

x_1
x_2
G_1
x_{11}
x_3
x_4
G_2
x_{12}
G_3
x_5
x_6
G_4
x_{15}
x_7
x_{13}
G_5
x_{16}
G_8
x_{19}
x_{14}
x_8
G_6
x_{17}
x_9
x_{10}
G_7
x_{18}
G_9
x_{20}
G_{10}
x_{21}

Figure 2. Network for Example 2.

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{18}
t_1	0		0													$\cancel{1}$		
					0								0					$\cancel{1}$
t_2		0		0													$\cancel{1}$	
										0				0	$\cancel{1}$			
t_3					$\cancel{2}$	$\cancel{2}$					0		$\cancel{1}$					
																0		
								$\cancel{1}$										0
t_4									$\cancel{2}$	$\cancel{2}$		0		$\cancel{1}$				
																	0	
							$\cancel{1}$								0			
t_5						0	0											
t_6								0	0									

Table 2. Multiple Fault Detection Table of the Test Set Derived for Example 2.

HAZARD ELIMINATION IN TWO-VALUED CIRCUITS BY MULTIVALUED TECHNIQUES*

EDGAR DUCASSE
Department of Computer and Information Science
Brooklyn College, CUNY
Brooklyn, New York 11210

ABSTRACT

This paper presents a unified approach to the construction of hazard-free implementations of Boolean functions by multivalued gates, making possible a cost analysis inclusive of all such types of realizations. The reliability of various such networks is also considered, and trade-offs between cost and reliability are discussed.

I. INTRODUCTION

Herein we consider several methods of solving a common problem in switching theory, the static and dynamic hazard-free realization of functions defined on Boolean algebras, by employing operators defined on a generalization of these algebras called Post lattices [6]. Our notation will be that employed in this reference. The Post functions of interest in this paper are those which have implementations that produce no temporary false signal values. For details of these functions which have no non-adjacent values for adjacent arguments and are called operation-hazard-free or O-H-F, see either [4] or [5]. The process by which a Boolean function can be expressed in a hazard-free form using Post functions can also be found in these references, as can a discussion of the n-tuple representation of a Post function which will be used in the sequel. Employing this process one obtains a unique representation of a given Boolean operator in terms of Post functions, all of whose values are 0 or 1. In the next section it will be shown how additional representations can be obtained. We will then investigate how all the Post functions employed in these realizations can be constructed using monotone functions which have only a single change in signal value. Finally, the costs of these various hazard-free implementations of a Boolean operator will be compared.

---

[*] An expanded version of this paper will appear as a monograph in a forthcoming book on multiple-valued logic and its applications in computer science.

II. COST COMPARISONS

Those functions, each of whose values are 0 or 1, which are used in the representation of a Boolean operator, are operation-hazard-free. An important property of this set of functions is that any k-variable operator defined on a finite Boolean algebra $B(2^n)$ can be expressed in terms of them. Note that there are always other collections of O-H-F functions in $G[P(n):x]$, each member of which assumes at most two adjacent values. In $G[P(3):x]$, for example, there is the class $\{111,112,121,122,211,212,221,222\}$.

An important question is whether or not these other classes in $G[P(2n):x]$ can be employed to realize Boolean operators in the same sense as the functions whose values are 0 and 1. It should be clear from the n-tuple representation $g = g_o \ldots g_{n-1}$, where $g_j = g(j)$, the j^{th} element of the n-chain $P(n)$, that the set of all one-variable functions defined on $P(n)$, each of which assumes only the values i or i+1, $0 \leq i \leq n-2$, is precisely the interval $[e_i, e_{i+1}]$. To show that the elements of $[e_i, e_{i+1}]$, $i \geq 1$, in $G[P(2n):x]$ are sufficient to realize any Boolean operator on $B(2^n)$ it suffices to prove that the function $\varphi(x) = e_i + e_{i+1} \cdot C_o^2(x)$ defined on $[\theta, e_1]$ is an isomorphism of this interval onto $[e_i, e_{i+1}]$. For details of this proof see [4]. The sets of mappings $[e_i, e_{i+1}]$ in $G[P(2n):x]$ are thus seen to be distinguished classes of this lattice. Indeed, they possess additional desirable properties. For a complete discussion see [4,5].

If we wish to be able to realize any k-variable Boolean operator on $B(2^n)$ in a form free of static and dynamic hazards, it will be necessary to have available all the members of some interval $[e_i, e_{i+1}]$ of the Post algebra $G[P(2n):x_i]$. This implies that a cost comparison should be carried out for entire intervals of O-H-F Post functions. Fortunately, all these operators can be put into an easily recognizable standard form from which cost comparisons can be made. Our cost analysis will be carried out in an arbitrary lattice $G[P(n):x] \cong [P(n)]^n$. Let $f(x) \in G[P(n):x]$ and assume f lies in the interval $[e_i, e_{i+1}]$. Our cost criterion will be the total number of inputs to logic elements of a network, and it will be assumed that settlement operators and constant functions are not available as inputs, but cost one lead each. If ℓ is the number of occurrences of i in the n-tuple representation of f, then the cost of an implementation of f is the same as one of the n-tuple f_1 having i for its first ℓ components and i+1 for the other $n-\ell$. If $1 \leq \ell \leq [\frac{n-1}{2}]$, the standard form of f_1 is

$$i + (i+1)\cdot C_o(\sum_{j=0}^{\ell-1} C_j(x)). \qquad (a)$$

If $[\frac{n-1}{2}]+1 \leq \ell \leq n-1$, the standard form of this function is

$$i + (i+1)\cdot \sum_{j=\ell}^{n-1} C_j(x). \qquad (b)$$

If $\ell = 0$ then f_1 (and also f) is the constant function i+1, whereas if $\ell = n$ then $f_1(x) = f(x) = i$ for all $x \in P(n)$. On the basis of these forms we can derive cost expressions (α), (β), and (γ) for the intervals of functions $[e_i, e_{i+1}]$, $0 < i < n-2$, $[e_o, e_1]$, and $[e_{n-2}, e_{n-1}]$ respectively. The details may be found in [5]. The accompanying table shows the cost of the sets $[e_i, e_{i+1}]$ in $G[P(n):x]$ for small values of n, and the difference in cost of the two least expensive intervals.

n	(α)	(β)	(γ)	(β)-(γ)
3	-	29	29	0
4	122	80	74	6
5	287	197	187	10
6	647	461	426	35
7	1423	1045	989	56
8	3090	2328	2174	154
9	6647	5117	4871	246
10	14,219	11,153	10,526	627

Table 1

The cost analysis for hazard-free realizations of Boolean operators carried out above considers realizations in terms of the functions C_i. However, these functions are difficult to implement physically. This is due to the fact that they cannot be implemented by single-threshold devices. The following figure detailing the characteristics of such a function demonstrates why this is the case.

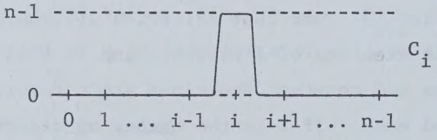


Figure 1

The operators we introduce now, on the other hand, present no such problem since they can be implemented by single-threshold devices. For $i=1,\ldots,2n-2$ we define functions $g_i(x)$ on $P(n)$ by $g_i(x) = \underbrace{(n-1)\ldots(n-1)}_{i}\underbrace{0\ldots0}_{n-1}$ if $1 \leq i \leq n-1$, and $g_i(x) = \underbrace{0\ldots0}_{i-n+1}\underbrace{(n-1)\ldots(n-1)}_{2n-i-1}$ if $n \leq i \leq 2n-2$. The behavior of these functions is shown in the figure below.

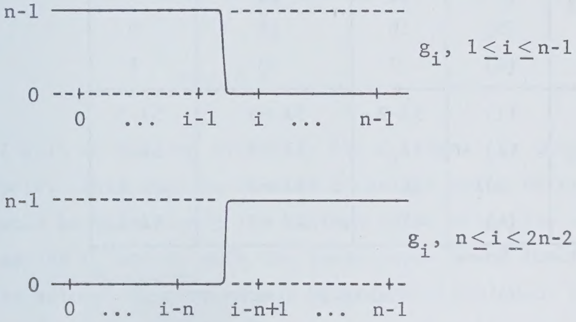


Figure 2

All of these 2n-2 functions are either monotone increasing or decreasing and each has a single change in signal value. It is known [11] that this set of operators generates every Post function on $P(n)$. We will now carry out a second cost analysis for hazard-free realizations of Boolean operators, this time considering realizations constructed using these single-threshold functions. Our cost criterion will be identical to that employed in the first analysis.

Each function $g \in [e_i, e_{i+1}]$ can be regarded as being a finite sequence of blocks of i's and (i+1)'s. Assume the number of blocks containing the initial value in the n-tuple representation $g = g_0 \ldots g_{n-1}$ is ℓ. We can derive a standard form for g depending solely on its block structure. There are four distinct cases, the details of which can be found in [5]. Also found there are details of the cost analysis which is summarized in Table 2. It can be seen from this table that every function has a cheapest realization in $[e_{n-2}, U]$, except that those from $[\theta, e_1]$ in case (1) for $\ell = 2$ have one less input lead than those from $[e_{n-2}, U]$. We therefore conclude that the interval $[e_i, e_{i+1}]$ of Post functions from $G[P(2n):x]$ which should be chosen for the hazard-free implementation of Boolean operators is $[e_{n-2}, U]$. This is the same conclusion reached in

ℓ	Case	$i=0$	$0<i<n-2$	$i=n-2$
$\ell=1$	(1)	1	1	1
	(2)	4	7	4
	(3)	4	7	4
	(3)	1	1	1
$\ell=2$	(1)	6	9	7
	(2)	10	13	9
	(3)	10	13	9
	(4)	7	10	6
$\ell\geq 3$	(1)	$5\ell-2$	$5\ell+1$	$5\ell-3$
	(2)	5ℓ	$5\ell+3$	$5\ell-1$
	(3)	5ℓ	$5\ell+3$	$5\ell-1$
	(4)	4ℓ	$4\ell+3$	$4\ell-1$

Table 2

the first cost analysis. To conclude this paper, we now compare the two classes of hazard-free implementations studied above, consider the relative costs of these classes, and discuss trade-offs in cost and reliability.

To best facilitate comparison of these two classes of hazard-free implementations of Boolean functions we should consider networks designed using the same class of operators. In the first cost analysis the C_i functions were used, while for the second we employed monotone functions which could be implemented by single-threshold devices. The question naturally arises whether it is possible to construct one of these classes of functions from the other. This would save us some trouble since if this were possible, we would have to redesign only one of the two classes of hazard-free implementations instead of both. We consider first the possibility of constructing the single-threshold functions g_i from the operators $C_o,\ldots,C_{n-1}$. It seems plausible that each of these functions should be realizable as a least upper bound of the C_i. For example, it first appears that the function g_2 of the Post algebra of functions $[P(3)]^3$ will perform correctly when implemented as the least upper bound C_o+C_1. To see that this may not be the case, consider the following figure.

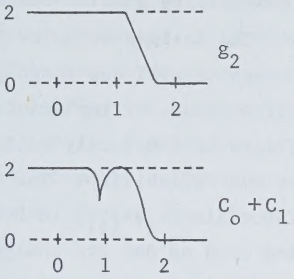


Figure 3

The C_o signal will be falling off while the C_1 signal is changing to its higher threshold. This implies that at a certain point in time, the function g_2 will be maintaining its maximum value while the signal produced by gating C_o and C_1 with the least upper bound function will be less than this value. Thus we cannot effectively implement the single-threshold functions g_i using the disjoint set of operators $C_o, \ldots, C_{n-1}$.

We turn next to the possibility of reliably constructing the operators C_i from the set of functions g_i. Such implementations are not only feasible, but are easily constructed. An operator which is not itself a single-threshold function (C_o and C_{n-1} are themselves such functions) can be realized by simply gating two of the g_i functions with the greatest lower bound operator. Consider the following example showing the C_1 operator of $[P(3)]^3$ being implemented in a form free from temporary false signal values by gating the functions g_2 and g_3 with the g.l.b. operator.

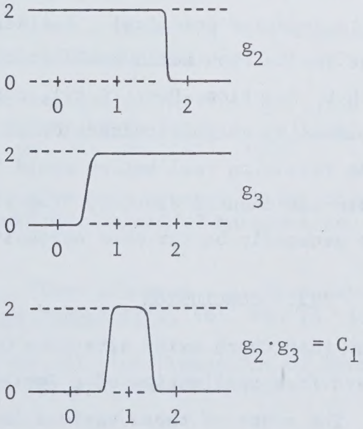


Figure 4

These last two examples demonstrate some of the merits and demerits of our two cost analyses. The designs employed in the second analysis are highly reliable, although we will see shortly that they are more costly when compared to the first class of implementations. The first class on the other hand may not always be completely reliable. Thus we will have a trade-off between cost and reliability. The following table shows the relative costs of the intervals $[e_i, e_{i+1}]$ in Post algebras $[P(n)]^n$ for small values of n using each of our two analyses.

n	Analysis	$i = 0$	$0 < i < n-2$	$i = n-2$
3	A1 A2	29 31	- -	29 31
4	A1 A2	80 85	122 127	74 83
5	A1 A2	197 217	287 307	187 207

Table 3

It is clear from the table that the designs employed in the second analysis are somewhat more costly than those in the first. The choice then is between cost and reliability.

Now let us attempt to improve the reliability of the first class of hazard-free realizations by implementing each C_i function as the greatest lower bound of two single-threshold functions. Even a cursory glance is sufficient to convince us that this seemingly attractive alternative is far too costly to consider practical. Besides introducing additional levels into the realization which would create more signal delay and distortion, each C_i function, $0 < i < n-1$, costing one input lead would have to be replaced by an interconnection of gates costing a total of four inputs. The resulting realization would then not only be no more reliable than those implemented directly from single-threshold functions, but would also generally be far more expensive.

III. CONCLUSION

It has been shown that there exist alternate ways of obtaining a static and dynamic hazard-free realization of a Boolean operator in terms of Post functions. The costs of these various implementations have been evaluated and compared, and one particular set of multivalued functions has been found to generally produce the least expensive hazard-free realizations. This was shown to be the case in both cost analyses. Comparisons were then carried out between the two classes of realizations considered in the cost analyses and those constructed using single-threshold devices were found to be more reliable and slightly more expensive than those implemented from the C_i functions. An attempt to raise the reliability of this latter set of realizations by constructing the C_i functions themselves from single-threshold devices was determined to be very costly.

REFERENCES

1. Birkhoff, G., Lattice Theory, American Mathematical Society Colloquium Publication 25, revised edition, New York, 1948.

2. DuCasse, E. and Metze, G., "Some Results on Computing Function Values in Finite Post Algebras," Coordinated Science Laboratory Report R-539, University of Illinois, Urbana, 1971.

3. DuCasse, E. and Metze, G., "The Relation of Equivalence for Post Algebras," Proceedings of the Tenth Annual Allerton Conference on Circuit and System Theory, Urbana, Illinois, pp. 625-634, October, 1972.

4. DuCasse, E. and Metze, G., "Hazard-Free Realizations of Boolean Functions Using Post Functions," Conference Record of the 1973 Symposium on the Theory and Applications of Multiple-Valued Logic Design, Toronto, Canada, pp. 59-67, May, 1973.

5. DuCasse, E. and Metze, G., "A Cost Analysis of Hazard-Free Networks Synthesized from Monotone Jump Functions," Proceedings of the Eleventh Annual Allerton Conference on Circuit and System Theory, Urbana, Illinois, pp. 400-407, October, 1973.

6. Epstein, G., "The Lattice Theory of Post Algebras," Transactions of the American Mathematical Society, Vol. 95, pp. 300-317, 1960.

7. Metze, G., "An Application of Multivalued Logic Systems to Circuits," Proceedings of a Symposium on Circuit Analysis, University of Illinois, Urbana, pp. 11-1 to 11-14, 1955.

8. Post, E., "Introduction to a General Theory of Elementary Propositions," American Journal of Mathematics, Vol. 43, pp. 163-185, 1921.

9. Rosenbloom, P. C., "Post Algebras. I. Postulates and General Theory," American Journal of Mathematics, Vol. 64, pp. 167-188, 1942.

10. Trazcyk, T., "Axioms and Some Properties of Post Algebras," Colloquium Mathematicum, Vol. 10, pp. 193-209, 1963.

11. Wojcik, A. S., "Relationships Between Post and Boolean Algebras with Application to Multivalued Switching Theory," Coordinated Science Laboratory Report R-512, University of Illinois, Urbana, 1971.

12. Wojcik, A. S. and Metze, G., "Some Relationships Between Post and Boolean Algebras," Conference Record of the 1971 Symposium on the Theory and Applications of Multiple-Valued Logic Design, Buffalo, New York, pp. 173-182, May, 1971.

THE NUMBER OF HAZARD-FREE MULTIPLE-VALUED FUNCTIONS

EDGAR DUCASSE
Department of Computer and Information Science
Brooklyn College, CUNY
Brooklyn, New York 11210

ABSTRACT

The number of m^{m^k} of m-valued functions of k variables, where $m > 2$, far exceeds the number of switching functions, 2^{2^k}. However, many of these multivalued functions possess characteristics which prevent them from being implementable in a hazard-free form. In this paper we enumerate the multivalued functions which can be implemented in a form free of static and dynamic hazards.

I. INTRODUCTION

In this section we will outline the development of a result which will enable us to distinguish multivalued functions which can be implemented in a hazard-free form from those which cannot. In the following section we will employ this criterion to enumerate the functions having realizations which do not cause hazardous operation. The multiple-valued operators we consider in the sequel are defined on a generalization of Boolean algebra called Post lattices [4]. Finite Post lattices are known [7] to be direct products $[P(n)]^m$ of n-chains for $n \geq 2$. This implies that elements of these lattices can be represented by m-tuples, each of whose components lies between 0 and n-1.

The importance of this representation theory is that it provides a means of representing functions defined on these lattices, also. If $g(x_1,\ldots,x_k)$ is a k-variable function defined on a Post algebra P(n), then g can be represented as a k-dimensional configuration of values between 0 and n-1 having n values on a side. Thus in the one-dimensional case, which will be considered first, $g(x_1)$ is representable as an n-tuple $g_o \cdots g_{n-1}$, where $g_i = g(i)$ for all $i = 0,\ldots,n-1$.

Now if 0,1,...,n-1 represent a set of n linearly ordered signal values employed in an asynchronous circuit, then a transition between any two nonadjacent values produces a temporary false output from the network. This is called an operation hazard. This forces us to reject for use in asynchronous circuits any multivalued function having non-adjacent outputs for adjacent inputs. In fact, it can be shown [3] that Post operators having no non-adjacent values for adjacent arguments, called operation-hazard-free or OHF, are precisely those that have implementations which are free from static and dynamic hazards.

To see that this is indeed the case, consider the following design process.

Design Rule. Let u and v be adjacent argument vectors of an OHF function $g(x_1,\ldots,x_k)$. The representation of g which is implemented by an asynchronous network is the one having the following characteristics.

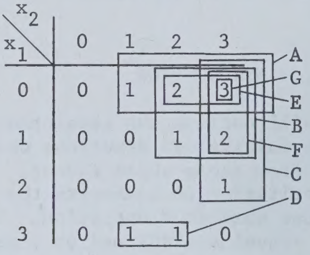
(a) If $g(u) = i \neq 0$, then g has at least one term that takes on the value i for the argument vector u.

(b) If $g(u) = i$ and $g(v) = j$ with $i \leq j$, then g contains a single term that takes on the value i for both arguments u and v.

(c) g has terms that take on the value i, $1 \leq i \leq n-1$, for the largest possible set of adjacent argument vectors, all of whose functional values are no smaller than i.

(d) If $g(u) = i \neq 0$ and u is included in a term of g that cannot assume a value greater than i-1, then u must be included in at least one term of g that can take on values no greater than i.

Let us consider an example of an application of the Design Rule. Table 1 defines a 2-variable OHF Post function on $[P(4)]^2$.



$x_1 \backslash x_2$	0	1	2	3
0	0	1	2	3
1	0	0	1	2
2	0	0	0	1
3	0	1	1	0

Table 1. A 2-variable OHF function

The representation for g obtained using the Design Rule is

$$g(x_1,x_2) = \underbrace{1\cdot(3000)_1\cdot(0333)_2}_{A} + \underbrace{1\cdot(3300)_1\cdot(0333)_2}_{B}$$

$$+ \underbrace{1\cdot(3330)_1\cdot(0003)_2}_{C} + \underbrace{1\cdot(0003)_1\cdot(0330)_2}_{D}$$

$$+ \underbrace{2\cdot(3000)_1\cdot(0033)_2}_{E} + \underbrace{2\cdot(3300)_1\cdot(0003)_2}_{F}$$

$$+ \underbrace{(3000)_1\cdot(0003)_2}_{G}$$

where the terms marked with letters correspond to the "loops" in the table and the subscripts on the 4-tuples correspond to the subscripts on the variables x_1 and x_2.

Now assume that an OHF function has been realized using the Design Rule and that the current argument vector u is changed to an adjacent argument v, where $v \neq u$. Assume also that gate and line delays of the circuit are finite but otherwise arbitrary, and that changes between adjacent arguments are made only after the network has completely responded to the previous input changes. We can then establish the following result.

Theorem 5. The realization of any k-variable OHF function resulting from the Design Rule is free of static and dynamic hazards.

We now know that the functions we wish to enumerate are precisely the OHF Post functions.

II. ONE-VARIABLE FUNCTIONS

In this section we present two techniques for enumerating one-variable OHF Post functions. The first method yields very explicit solutions, but is long and tedious to calculate, particularly if the number of values n is large. Since $g(x_1):P(n) \rightarrow P(n)$ can be represented as $g_0 \ldots g_{n-1}$, it follows that g is OHF if $|g_i - g_{i+1}| \leq 1$ for all $i = 0,\ldots,n-2$. Now let $0_n, 1_n, 2_n, \ldots$ denote the number of n-tuples ending in $0,1,2,\ldots$ respectively. Then

$$\begin{aligned} 0_n &= 0_{n-1} + 1_{n-1} \\ 1_n &= 0_{n-1} + 1_{n-1} + 2_{n-1} \\ &\vdots \\ (m-1)_n &= (m-2)_{n-1} + (m-1)_{n-1} + m_{n-1} \\ m_n &= (m-1)_{n-1} + m_{n-1} . \end{aligned}$$

We are thus considering a generalization of the problem originally posed, which is the degenerate case $m = n$. The following explicit formulas for small values of m were obtained from routinely calculated recurrence relations

$$m = 3: \frac{1}{2}\left[(1+\sqrt{2})^{n+1} + (1-\sqrt{2})^{n+1}\right]$$

$$m = 4: \left(1 + \frac{1}{\sqrt{5}}\right)\left(\frac{3+\sqrt{5}}{2}\right)^n + \left(1 - \frac{1}{\sqrt{5}}\right)\left(\frac{3-\sqrt{5}}{2}\right)^n$$

$$m = 5: \frac{1}{3} + \left(\frac{2+\sqrt{3}}{6}\right)(1+\sqrt{3})^{n+1} + \left(\frac{2-\sqrt{3}}{6}\right)(1-\sqrt{3})^{n+1}.$$

Substituting for n we obtain the following table.

m	BINARY	TERNARY (OHF)	TERNARY	4-VALUED (OHF)	4-VALUED	5-VALUED (OHF)	5-VALUED
3	8	17	27	26	64	35	125
4	16	41	81	68	256	95	625
5	32	99	243	178	1,024	259	3,125
6	64	239	729	466	4,096	707	15,625
7	128	577	2,187	1,220	16,384	1,931	78,125
8	256	1,393	6,561	3,194	65,536	5,275	390,625
9	512	3,363	19,683	8,362	262,144	14,411	1,953,125
10	1,024	8,119	59,049	21,892	1,048,576	39,371	9,765,625

Table 1. The number of n-Valued OHF Functions for $n = 3,4,5$

From the table it can be seen that the number of 3,4 and 5-valued OHF functions are 17, 81 and 259 respectively. Although less than the total number 3^3, 4^4 and 5^5 of 3, 4 and 5-valued functions, they considerably exceed the number of binary functions, 2^2, thus allowing the logic designer greater flexibility in his choice of gates.

The table above can also be obtained by an application of another technique [1] summarized here. If $g_k(n,a)$ denotes the number of sequences $a_1 \ldots a_n$ that satisfy $|a_i - a_{i+1}| \leq 1$ for all $i = 1,\ldots,n-1$, $0 \leq a_i \leq a$ for all $i = 1,\ldots,n$, and have $a_1 = k$, then $g_o(n,a) = \sum_{j=0}^{k} A'_{kj} g_{k-j}(n-k,a)$, $0 \leq k \leq a$, and $g_k(n,a) = \sum_{j=0}^{k} B'_{kj} g_o(n+k-j,a)$, $0 \leq k \leq a$, where $A'_{kj} = c(k,k+j) - c(k,k+j+2)$, $0 \leq j \leq k$, and $B'_{kj} = c(-k+j-1,j)$, $0 \leq j \leq k$, while $c(n,j)$ is defined by $(1+x+x^2) = \sum_{j=0}^{\infty} c(n,j)x^j$. This has the advantage of including all cases in one set of equations, although explicit solutions are tedious to obtain.

III. k-VARIABLE FUNCTIONS

It would be convenient if techniques for the one-variable case were directly extendable to k-variable functions. Unfortunately, this does not appear to be so. For consider the problem of reducing the 2-variable case to that of one variable. If one attempts to enumerate one-dimensional "lines," in the 2-dimensional "plane" representing the function, he immediately encounters the problem of adjacencies with neighboring lines. This problem becomes considerably more involved as the number of variables increases. Thus it appears that a non-trivial extension of present techniques is necessary to conquer the general case.

IV. CONCLUSION

A criterion was presented for determining multivalued functions having implementations free from static and dynamic hazards. Based on this criterion an enumeration of one-variable functions was given which demonstrates that the number of such functions significantly exceeds the corresponding number of binary functions. An alternative enumeration was then considered. Finally, the general case of k-variable multivalued functions was studied briefly and it appears that methods employed in the one-variable case are not readily extendable.

REFERENCES

1. Carlitz, L.,"Enumeration of Certain Sequences," Acta Arithmetica, Vol. XVIII, 1971, pp. 221-232.

2. DuCasse, E. and Metze, G., "Hard-Free Realizations of Boolean Functions Using Post Functions," Conference Record of the 1973 Symposium on the Theory and Applications of Multiple-Valued Logic Design, Toronto, Canada, May, 1973, pp. 59-67.

3. DuCasse, E., Wojcik, A., and Metze, G., "Cost Comparisons of Hazard-Free Realizations of Boolean Operators Synthesized From Multi-Valued Functions," in preparation.

4. Epstein, G., "The Lattice Theory of Post Algebras," Transactions of the American Mathematical Society, Vol. 95, pp. 300-317, 1960.

5. Metze, G. A., "An Application of Multivalued Logic Systems to Circuits," Proceedings of a Symposium on Circuit Analysis, University of Illinois, Urbana, pp. 11-1 to 11-14, 1955.

6. Post, E., "Introduction to a Theory of Elementary Propositions," American Journal of Mathematics, Vol. 43, pp. 163-185, 1921.

7. Rosenbloom, P. C., "Post Algebras, I. Postulates and General Theory," American Journal of Mathematics, Vol. 64, pp. 167-188, 1942.

8. Trazcyk, T., "Axioms and Some Properties of Post Algebras," Colloquium Mathematicum, Vol. 10, pp. 193-209, 1963.

9. Wojcik, A. S. and Metze, G., "Some Relationships Between Post and Boolean Algebras," Conference Record of the 1971 Symposium on the Theory and Applications of Multiple-Valued Logic Design, Buffalo, New York, pp. 173-182, May, 1971.

ON TESSELLATION AUTOMATA CAPABLE OF MIXED PARALLEL-SEQUENTIAL PROCESSING*

WILLIAM I. GROSKY
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, Georgia

FRANK TSUI
IBM Corporation
Gaithersburg, Maryland

ABSTRACT

We consider the equivalence of mixed parallel-sequential local operations of arbitrary neighborhood index in arbitrary dimensions, where the local operation and neighborhood index are the same for each mode of processing. We then relax this latter condition and explore the notion of simulation in general.

INTRODUCTION

Several recent research efforts are aimed at strengthening the theoretical understanding of parallel and sequential modes of picture and pattern processing [1,2,3,6,7,8,13]. Rosenfeld and Pfaltz [11] have shown that any picture transformation that can be accomplished by a series of parallel local operations with Moore neighborhood index can also be accomplished by a series of sequential local operations with Moore neighborhood index, and conversely; but the local operations may be different for the two types of processing. Rosenfeld and Milgram [9,10] continue this study.

This paper is a continuation of Grosky and Tsui [4,5]. We first concentrate our attention on the equivalence of mixed parallel-sequential local operations of arbitrary neighborhood index in arbitrary dimensions, where the local operation and neighborhood index are the same for each mode of processing. We then relax this latter condition and explore the notion of simulation in general. The methodology is that of tessellation automata and various generalizations of them.

STRATIFIED TESSELLATION AUTOMATA

DEFINITION 1: For $n \geq 1$, an n-dimensional stratified tessellation automoton, STA, is a 5-tuple $\langle S, Z^n, NI, LT, TR \rangle$, where,

1) S is a finite, non-empty set of states

2) Z^n is the set of n-tuples of integers. For $\underline{\zeta} \in Z^n$, we call $\underline{\zeta}$ a cell of STA. The set $CON = \{c \mid c: Z^n \to S\}$ is called the set of configurations of STA.

3) NI is an ordered q-tuple of elements of Z^n, for some finite $q \geq 1$, and is called the neighborhood index of STA. Suppose $NI = \langle \underline{\gamma}_1, \ldots, \underline{\gamma}_q \rangle$. Then, for $\underline{\zeta} \in Z^n$, $Ne(\underline{\zeta}) = \langle \underline{\zeta} + \underline{\gamma}_1, \ldots, \underline{\zeta} + \underline{\gamma}_q \rangle$ is called the neighborhood of $\underline{\zeta}$.

4) $LT = \langle \sigma_1, \ldots, \sigma_p \rangle$ for some finite $p \geq 1$, where, for $1 \leq i \leq p$, $\sigma_i \in \{\theta \mid \theta : S^q \to S\}$, the set of local transformations of STA

5) $TR = \langle \tau_1, \ldots, \tau_p \rangle$, where p is as in 4). For each $1 \leq i \leq p$, τ_i is called a trajectory, and is a member of $\{\phi \mid \phi : \{0,1,2,\ldots\} \to \mathcal{P}(Z^n)\}$

---

* This work was supported in part by NSF Grant GN-655

with the stipulations that,

a) $k,m \geqslant 0,\ k \neq m \rightarrow \tau_i(k) \cap \tau_i(m) = \emptyset$

b) $\bigcup\limits_{j=0} \tau_i(j) = Z^n$

For $1 \leqslant i \leqslant p$, σ_i and τ_i determine an element $g^{(STA)}_{\sigma_i,\tau_i} \in CON^{CON}$, the set of global transformations, as follows,

Suppose $c \in CON$. Then $g^{(STA)}_{\sigma_i,\tau_i}(c) = \hat{c}$, where, for each cell $\underline{\zeta} \in Z^n$, we define $\hat{c}(\underline{\zeta})$ in the following fashion,

Let $\alpha_{\underline{\zeta}} \in \{0,1,2,\ldots\}$ be such that $\underline{\zeta} \in \tau_i(\alpha_{\underline{\zeta}})$. We now define $c_0^{(\underline{\zeta})},\ldots,c_{\alpha_{\underline{\zeta}}}^{(\underline{\zeta})} \in CON$:

$$c_0^{(\underline{\zeta})} = c$$

For $0 \leqslant k \leqslant \alpha_{\underline{\zeta}} - 1$ and $\underline{\xi} \in Z^n$,

$$c_{k+1}^{(\underline{\zeta})}(\underline{\xi}) = \begin{cases} \sigma_i(c_k^{(\underline{\zeta})}(\underline{\xi} + \underline{\gamma}_1),\ldots,c_k^{(\underline{\zeta})}(\underline{\xi} + \underline{\gamma}_q)), & \text{if } \underline{\xi} \in \tau_i(k) \\ c_k^{(\underline{\zeta})}(\underline{\xi}) & \text{otherwise} \end{cases}$$

Then, $\hat{c}(\underline{\zeta}) = c_{\alpha_{\underline{\zeta}}}^{(\underline{\zeta})}(\underline{\zeta})$

The trajectory indicates the sequential order in which clumps of cells of Z^n are processed. Cells in the same clump are processed in parallel. Notice that $\tau_i(k)$, for some $k \geqslant 0$, could be $\emptyset$. This indicates that no processing is done in the given time period.

τ_i is a trajectory for pure sequential processing if, for each $k \geqslant 0$, $|\tau_i(k)| \leqslant 1$, while it is a trajectory for pure parallel processing if there is some $k \geqslant 0$ such that $\tau_i(k) = Z^n$.

DEFINITION 2: For $\sigma: S^q \rightarrow S$, if σ is independent of its j-th argument, for $1 \leqslant j \leqslant q$, we call cell $\underline{\zeta} + \underline{\gamma}_j$ an independent neighbor of $\underline{\zeta}$ with respect to σ, for each $\underline{\zeta} \in Z^n$.

DEFINITION 3: Let $\underline{\zeta} \in Z^n$ and τ be a trajectory. Suppose $\alpha_{\underline{\zeta}} \in \{0,1,2,\ldots\}$ is such that $\underline{\zeta} \in \tau(\alpha_{\underline{\zeta}})$. Then, $(\tau(0) \cup \ldots \cup \tau(\alpha_{\underline{\zeta}} - 1)) \cap Ne(\underline{\zeta})$ is called the preprocessed set of $\underline{\zeta}$ with respect to τ. This is just the collection of neighbors of $\underline{\zeta}$ which are processed via τ before $\underline{\zeta}$.

DEFINITION 4: We call a local transformation $\sigma: S^q \rightarrow S$ surjective of degree k/q, for $0 \leqslant k \leqslant q$, if, by varying the values of any k arguments of σ, we can produce as output every element of S, regardless of the values of the other $q-k$ arguments. That is, letting $1 \leqslant i_1 < \ldots < i_k \leqslant q$, the function $\sigma\#: S^k \rightarrow S$ defined by $\sigma\#(x_{i_1},\ldots,x_{i_k}) = \sigma(y_1,\ldots,y_q)$, where, for $1 \leqslant j \leqslant q$, $y_j = x_{i_j}$ if $j \in \{i_1,\ldots,i_k\}$ while $y_j \in S$ is arbitrary otherwise, is onto.

DEFINITION 5: Cell $\underline{\zeta}$ is said to be a related neighbor of cell $\underline{\xi}$ if there exists a chain of cells $\underline{\delta}_1,\ldots,\underline{\delta}_m$, for $m \geqslant 2$, such that $\underline{\delta}_1 = \underline{\zeta}$, $\underline{\delta}_m = \underline{\xi}$, and, for $1 \leqslant i \leqslant m-1$, $\underline{\delta}_i \in Ne(\underline{\delta}_{i+1})$.

DEFINITION 6: $\Psi_1 \subseteq Z^n$ is said to be a *related set* of $\Psi_2 \subseteq Z^n$ with respect to a trajectory τ if there are numbers $i_1, i_2 \geqslant 0$ where $i_1 \neq i_2$, such that $\tau(i_1) = \Psi_1$, $\tau(i_2) = \Psi_2$, and there are cells $\underline{\zeta}_1 \in \Psi_1$, $\underline{\zeta}_2 \in \Psi_2$ such that $\underline{\zeta}_1$ is a related neighbor of $\underline{\zeta}_2$.

DEFINITION 7: The *skeleton* of a trajectory τ is defined as the set $\{\tau(0), \tau(1), \tau(2), \ldots\}$. Two trajectories are called *skeleton equivalent* if they have the same skeleton.

DEFINITION 8: A state $s \in S$ is called *quiescent* with respect to $\sigma : S^q \to S$, if $\sigma(s, \ldots, s) = s$. A configuration c is called *finite* with respect to $\sigma : S^q \to S$ if only a finite number of cells of c are non-quiescent with respect to σ. We assume that S has at most *one* quiescent state.

DEFINITION 9: Suppose $\underline{0}$ is the j-th element of the neighborhood index of STA, for $1 \leqslant j \leqslant q$. A state $s \in S$ is called *null* with respect to $\sigma : S^q \to S$ if, for any $s_1, \ldots, s_{j-1}, s_{j+1}, \ldots, s_q \in S$, $\sigma(s_1, \ldots, s_{j-1}, s, s_{j+1}, \ldots, s_q) = s$. A configuration c is called *fixed* with respect to $\sigma : S^q \to S$ if only a finite number of cells of c are non-null with respect to σ. A configuration c is called *fixed of degree* $m \geqslant 0$ with respect to $\sigma : S^q \to S$, if, whenever the absolute value of any coordinate of a cell $\underline{\zeta}$ is larger than m, then $c(\underline{\zeta})$ is null with respect to σ.

DEFINITION 10: For τ a trajectory over Z^n and $A \subseteq Z^n$, we define the *index set of A* with respect to τ as $\{j \mid j \geqslant 0 \text{ and } \tau(j) \cap A \neq \emptyset\}$. We denote it as $IS_\tau(A)$.

DEFINITION 11: For τ a trajectory over Z^n and $A \subseteq Z^n$, we define the *covering set of A* with respect to τ as $\bigcup_{j \in IS_\tau(A)} \tau(j)$. We denote it as $CS_\tau(A)$.

DEFINITION 12: Let $STA = \langle S, Z^n, NI, \langle \sigma_1, \ldots, \sigma_{p_1} \rangle, \langle \tau_1, \ldots, \tau_{p_1} \rangle \rangle$ and $STA^* = \langle S^*, Z^n, NI^*, \langle \sigma_1^*, \ldots, \sigma_{p_2}^* \rangle, \langle \tau_1^*, \ldots, \tau_{p_2}^* \rangle \rangle$ be two stratified tessellation automata. For $t, r \geqslant 1$, we say that *STA* simulates STA in t/r times real time* if there are effectively computable injective mappings $\Delta : CON \to CON^*$ and $\Gamma : \{\sigma_1, \ldots, \sigma_{p_1}\}^r \to \{\sigma_1^*, \ldots, \sigma_{p_2}^*\}^t$ such that for any $c \in CON$ and any $\langle \sigma_{i_1}, \ldots, \sigma_{i_r} \rangle \in \{\sigma_1, \ldots, \sigma_{p_1}\}^r$, there are trajectories $\hat{\tau}_1, \ldots, \hat{\tau}_r$, $\bar{\tau}_1, \ldots, \bar{\tau}_t$ such that,

$$\Delta(g^{(STA)}_{\sigma_{i_1}, \hat{\tau}_1}(\ldots(g^{(STA)}_{\sigma_{i_r}, \hat{\tau}_r}(c)\ldots)) = g^{(STA^*)}_{\sigma^*_{i_1}, \bar{\tau}_1}(\ldots g^{(STA^*)}_{\sigma^*_{i_t}, \bar{\tau}_t}(\Delta(c))\ldots)$$

where $\langle \sigma^*_{i_1}, \ldots, \sigma^*_{i_t} \rangle = \Gamma(\langle \sigma_{i_1}, \ldots, \sigma_{i_r} \rangle)$.

$\{\bar{\tau}_1, \ldots, \bar{\tau}_t\}$ is called the set of *simulating trajectories*, while $\{\hat{\tau}_1, \ldots, \hat{\tau}_r\}$ is called the set of *simulated trajectories*.

If $t = r = 1$ and Δ is the identity function, we call the simulation *strict*. If $t = r = 1$ and Γ is the identity function, we call the simulation *operational*. If $\hat{\tau}_1, \ldots, \hat{\tau}_r, \bar{\tau}_1, \ldots, \bar{\tau}_t$ are independent of c, we call the simulation *uniform*. If, for $1 \leqslant j \leqslant r$, $\hat{\tau}_j = \tau_{i_j}$, and, for $1 \leqslant k \leqslant t$, $\overline{\tau}_k = \tau^*_{i_k}$, we call the simulation *proper*.

We may talk of simulation over finite or fixed configurations. This is the case if our map Δ is only defined over these types of configurations

DEFINITION 13: Let $STA = \langle S, Z^n, \langle \underline{\gamma}_1, \ldots, \underline{\gamma}_q\rangle, \langle \sigma_1, \ldots, \sigma_p\rangle, \langle \tau_1, \ldots, \tau_p\rangle\rangle$. Suppose $\underline{\zeta} \in Z^n$ and $1 \leqslant i \leqslant p$. We define what we call the *type* of cell $\underline{\zeta}$ with respect to τ_i, denoted by $\text{type}^{(STA)}_{\underline{\zeta},\tau_i}$, as follows,

Let $S_0 = \{*\}$, and, for $j \geqslant 0$, let $S_{j+1} = S_j \cup S_j^q$. Let $\alpha_{\underline{\zeta}} \in \{0,1,2,\ldots\}$ be such that $\underline{\zeta} \in \tau_i(\alpha_{\underline{\zeta}})$. For $0 \leqslant k \leqslant \alpha_{\underline{\zeta}}$, we define $\varepsilon_k^{(\underline{\zeta})} : Z^n \to S_k$:

Now, $\varepsilon_0^{(\underline{\zeta})}$ is the constant function which maps Z^n onto $\{*\}$. For $0 \leqslant k \leqslant \alpha_{\underline{\zeta}} - 1$ and $\underline{\xi} \in Z^n$, we define,

$$\varepsilon_{k+1}^{(\underline{\zeta})}(\underline{\xi}) = \begin{cases} \langle \varepsilon_k^{(\underline{\zeta})}(\underline{\xi} + \underline{\gamma}_1), \ldots, \varepsilon_k^{(\underline{\zeta})}(\underline{\xi} + \underline{\zeta}_q)\rangle & \text{if } \underline{\xi} \in \tau_i(k) \\ \varepsilon_k^{(\underline{\zeta})}(\underline{\xi}) & \text{otherwise} \end{cases}$$

Then, $\text{type}^{(STA)}_{\underline{\zeta},\tau_i} = \varepsilon_{\alpha_{\underline{\zeta}}}^{(\underline{\zeta})}(\underline{\zeta})$.

SOME SIMULATION RESULTS

THEOREM 1: Suppose $STA = \langle S, Z^n, NI, \langle \sigma_1, \ldots, \sigma_{p_1}\rangle, \langle \tau_1, \ldots, \tau_{p_1}\rangle\rangle$ and $STA^* = \langle S, Z^n, NI, \langle \sigma^*_1, \ldots, \sigma^*_{p_2}\rangle, \langle \tau^*_1, \ldots, \tau^*_{p_2}\rangle\rangle$ are such that there is a function $\theta : \{1, \ldots, p_1\} \to \{1, \ldots, p_2\}$ such that, for $1 \leqslant j \leqslant p_1$, $\sigma_j = \sigma^*_{\theta(j)}$. For $1 \leqslant j \leqslant p_1$ and for all $\underline{\zeta} \in Z^n$, suppose that the preprocessed set of $\underline{\zeta}$ with respect to τ_j consists entirely of independent neighbors of $\underline{\zeta}$ with respect to σ_j and that the preprocessed set of $\underline{\zeta}$ with respect to $\tau^*_{\theta(j)}$ consists entirely of independent neighbors of $\underline{\zeta}$ with respect to $\sigma^*_{\theta(j)}$. Then, STA^* simulates STA in a strict, operational and proper manner.

PROOF OF THEOREM 1: It suffices to show, for $1 \leqslant i \leqslant p_1$ and $c \in CON$, that $g_{\sigma_i,\tau_i}(c) = g_{\sigma_i,\tau^*_{\theta(i)}}(c)$. This will follow if we can show that given any cell $\underline{\zeta} \in Z^n$ and its set of neighbors $Ne(\underline{\zeta})$ in any states whatever, at a particular time t, the state of $\underline{\zeta}$ at time t+1 is the same whether we process cell $\underline{\zeta}$ under σ_i using trajectory τ_i or under $\sigma^*_{\theta(i)} = \sigma_i$ using trajectory $\tau^*_{\theta(i)}$. Now, by hypothesis, those cells of $Ne(\underline{\zeta})$ which are processed before $\underline{\zeta}$ using trajectories τ_i or $\tau^*_{\theta(i)}$ are independent neighbors of $\underline{\zeta}$ with respect to σ_i. Thus, the state of cell $\underline{\zeta}$ at time t+1 depends on the states of just those cell of $Ne(\underline{\zeta})$ which are not processed before $\underline{\zeta}$ using any of these two trajectories. These states are the same in both cases, and hence our result follows. QED

THEOREM 2: Suppose STA = $<S,Z^n,<\underline{Y}_1,\ldots,\underline{Y}_q>,<\sigma_1,\ldots,\sigma_{p_1}>,<\tau_1,\ldots,\tau_{p_1}>>$ and STA* = $<S,Z^n,<\underline{Y}_1,\ldots,\underline{Y}_q>,<\sigma^*_1,\ldots,\sigma^*_{p_2}>,<\tau^*_1,\ldots,\tau^*_{p_2}>>$ are such that STA* simulates STA in a strict, operational and proper manner. For some $1 \leq j \leq p_1$ and some cell $\underline{\zeta} \in Z^n$, suppose the following hold:

For some $\underline{\xi} \in$ PRE, the preprocessed set of $\underline{\zeta}$ with respect to τ_j, let $\beta_{\underline{\xi}} = Ne(\underline{\xi}) - (Ne(\underline{\xi}) \cap Ne(\underline{\zeta})) - (Ne(\underline{\xi}) \cap \bigcup_{\underline{\delta} \in PRE - \{\underline{\xi}\}} Ne(\underline{\delta}))$. Suppose there is some non-empty subset $SUB_{\underline{\xi}} \subseteq \beta_{\underline{\xi}}$ such that,

1) σ_j is surjective of degree $|SUB_{\underline{\xi}}|/q$

2) Every element of $SUB_{\underline{\xi}}$ is not in the preprocessed set of $\underline{\xi}$ with respect to τ_j

3) Let σ_j,τ be the local transformation and trajectory in STA* corresponding to σ_j,τ_j in STA via the simulation. Then, for all sets A and B, where $A \cap SUB_{\underline{\xi}} \neq \emptyset$ and $\underline{\zeta} \in B$, A is not a related set of B with respect to τ.

Then, $\underline{\xi} \in$ PRE is an independent neighbor of $\underline{\zeta}$ with respect to σ_j.

PROOF OF THEOREM 2: There are at least $\beta_{\underline{\xi}}$ cells in the neighborhood of $\underline{\xi}$, not including $\underline{\xi}$, which are neither in the neighborhood of $\underline{\zeta}$ nor in the neighborhood of any other cell in PRE.

Let $c_1,\ldots,c_{|S|}$ be $|S|$ configurations which differ only on the cells of $SUB_{\underline{\xi}}$ and which are such that the state of cell $\underline{\xi}$ in the successor configuration of c_k, $1 \leq k \leq |S|$, processed with respect to σ_j and τ_j, is the k-th state of S (with respect to some arbitrary ordering of S). This can be done by hypotheses 1) and 2) above. Considering the above initial configurations $c_1,\ldots,c_{|S|}$, when cell $\underline{\zeta}$ is processed as above, the only cell in its neighborhood which differs in state from one configuration to another is thus cell $\underline{\xi}$ and these states run through all the elements of set S.

Consider now the successor configurations of $c_1,\ldots,c_{|S|}$ when each are processed with respect to σ_j and τ. By hypothesis 3), no cell in $SUB_{\underline{\xi}}$ is a related neighbor to any cell in $Ne(\underline{\zeta}) \cup \{\underline{\zeta}\}$. Thus, the states of cell $\underline{\zeta}$ in each of these successor configurations must be the same.

Since the simulation is strict, operational and proper, we must also have that the states of cell $\underline{\zeta}$ in each of the successor configurations when processing is done with respect to σ_j and τ_j are the same. From the above, we immediately see that $\underline{\xi}$ is an independent neighbor of $\underline{\zeta}$ with respect to σ_j. QED

THEOREM 3: Suppose STA = $<S,Z^n,NI,<\sigma_1,\ldots,\sigma_{p_1}>,<\tau_1,\ldots,\tau_{p_1}>>$ and STA* = $<S,Z^n,NI,<\sigma^*_1,\ldots,\sigma^*_{p_2}>,<\tau^*_1,\ldots,\tau^*_{p_2}>>$ are such that there is a function $\theta:\{1,\ldots,p_1\} \to \{1,\ldots,p_2\}$ such that, for $1 \leq j \leq p_1$, $\sigma_j = \sigma^*_{\theta(j)}$. For $\tau \in \{\tau_1,\ldots,\tau_{p_1},\tau^*_{\theta(1)},\ldots,\tau^*_{\theta(p_1)}\}$, suppose that if A is a related set of B with respect to τ then B is not a related set of A with respect to τ. Then, there are two stratified tessellation automata, $\overline{STA} = \langle S,Z^n,NI,\langle\sigma_1,\ldots,\sigma_{p_1}\rangle,\langle\overline{\tau}_1,\ldots,\overline{\tau}_{p_1}\rangle\rangle$ and $\overline{STA^*} = \langle S,Z^n,NI,\langle\sigma^*_1,\ldots,\sigma^*_{p_2}\rangle,\langle\overline{\tau}^*_1,\ldots,\overline{\tau}^*_{p_2}\rangle\rangle$ such that,

1) For $1 \leqslant i \leqslant p_1$, τ_i and $\overline{\tau}_i$ are skeleton equivalent

2) For $1 \leqslant i \leqslant p_2$, $i \in \text{range}(\theta)$, τ^*_i and $\overline{\tau}^*_i$ are skeleton equivalent

3) $\overline{STA^*}$ simulates $\overline{STA}$ over the fixed configurations of degree $m \geqslant 0$, for a given m, in a strict, operational and proper manner

PROOF OF THEOREM 3: For each $\tau \in \{\tau_1,\ldots,\tau_{p_1},\tau^*_{\theta(1)},\ldots,\tau^*_{\theta(p_1)}\}$ we do the following:

Let $R_\tau = \{\langle A,B\rangle \mid A,B \subseteq Z^n$ and either $A = B$ or A is a related set of B with respect to $\tau\}$. It is easily seen that R_τ is a reflexive, transitive binary relation over $\mathcal{P}(Z^n)$. We now show that R_τ is also anti-symmetric. Suppose $\langle S_1,S_2\rangle \in R_\tau$ and $\langle S_2,S_1\rangle \in R_\tau$. Thus, either $S_1 = S_2$ or S_1 and S_2 are related sets of each other with respect to τ. Since the latter condition is false by hypothesis, we have our result. R_τ is thus a partial order over $\mathcal{P}(Z^n)$.

Let $W_m \subseteq Z^n$ be such that $\underline{\alpha} \in W_m$ iff the absolute value of each coordinate of $\underline{\alpha}$ is less than or equal to m. Let $S_\tau = CS_\tau(W_m)$. Now, from the above, we have that $R_\tau \cap (\mathcal{P}(S_\tau) \times \mathcal{P}(S_\tau))$ is a partial order over $\mathcal{P}(S_\tau)$. We embed this partial order in a total order T_τ. Thus, for all $i,j \in IS_\tau(W_m)$, $\tau(i)\ R_\tau\ \tau(j)$ implies that $\tau(i)\ T_\tau\ \tau(j)$.

Let $\{\psi_1,\ldots,\psi_\ell\}$ be a complete listing of $\{\tau(i) \mid i \in IS_\tau(W_m)\}$, such that, if $1 \leqslant r < s \leqslant |IS_\tau(W_m)|$ then $\tau(r)\ T_\tau\ \tau(s)$. We now define a trajectory $\hat{\tau}$ over Z^n with the same skeleton set as τ but which is such that, if, for $0 \leqslant r < s$, $\hat{\tau}(r) = \psi_{i_r}$ and $\hat{\tau}(s) = \psi_{i_s}$, where $1 \leqslant i_r, i_s \leqslant \ell$, then $i_s < i_r$. ($\hat{\tau}$ may be the same as τ except that ℓ entries are shifted around in some other order.)

For c any configuration and σ the local transformation associated with τ, it is straightforward to see that the successor configuration of c processed with respect to σ and $\hat{\tau}$ is the same as the successor configuration of c processed with respect to σ and a trajectory for pure parallel processing.

Letting $\overline{\tau}_1 = \hat{\tau}_1,\ldots,\overline{\tau}_{p_1} = \hat{\tau}_{p_1}$, $\overline{\tau}^*_{\theta(1)} = \hat{\tau}^*_{\theta(1)},\ldots,\overline{\tau}^*_{\theta(p_1)} = \hat{\tau}^*_{\theta(p_1)}$ and, for $j \notin \text{range}(\theta)$, letting $\overline{\tau}^*_j$ be arbitrary, our result easily follows. QED

COROLLARY 1: Suppose $STA = \langle S,Z^n,NI,\langle\sigma_1,\ldots,\sigma_{p_1}\rangle,\langle\tau_1,\ldots,\tau_{p_1}\rangle\rangle$ and $STA^* = \langle S,Z^n,NI,\langle\sigma^*_1,\ldots,\sigma^*_{p_2}\rangle,\langle\tau^*_1,\ldots,\tau^*_{p_2}\rangle\rangle$ are such that there is a function $\theta:\{1,\ldots,p_1\} \to \{1,\ldots,p_2\}$ such that, for $1 \leqslant j \leqslant p_1$, $\sigma_j = \sigma^*_{\theta(j)}$. For $\tau \in \{\tau_1,\ldots,\tau_{p_1},\tau^*_{\theta(1)},\ldots,\tau^*_{\theta(p_1)}\}$, suppose that if A is a related set of B with respect to τ then B is not a related set of A with respect to τ. With R_τ and S_τ as in the previous proof, also assume that for $i,j \geqslant 0$, $i \neq j$, if $\tau(i)\ R_\tau \cap (\mathcal{P}(S_\tau) \times \mathcal{P}(S_\tau))\ \tau(j)$ then $j < i$. Then STA* simulates

STA over the fixed configurations of degree $m \geqslant 0$ in a strict, operational and proper manner.

PROOF OF COROLLARY 1: This follows directly from Theorem 3. QED

THEOREM 4: Suppose that $STA = \langle S, Z^n, NI, \langle \sigma_1, \ldots, \sigma_{p_1} \rangle, \langle \tau_1, \ldots, \tau_{p_1} \rangle\rangle$ and $STA^* = \langle S, Z^n, NI, \langle \sigma_1^*, \ldots, \sigma_{p_2}^* \rangle, \langle \tau_1^*, \ldots, \tau_{p_2}^* \rangle\rangle$ are such that there is a function $\theta : \{1, \ldots, p_1\} \to \{1, \ldots, p_2\}$ such that, for $1 \leqslant j \leqslant p_1$, $\sigma_j = \sigma_{\theta(j)}^*$. For $\tau \in \{\tau_1, \ldots, \tau_{p_1}, \tau_{\theta(1)}^*, \ldots, \tau_{\theta(p_1)}^*\}$, suppose that if A is a related set of B with respect to τ then B is not a related set of A with respect to τ. Then, we have that,

1) STA* simulates STA over the finite configurations in a strict and operational manner

2) Suppose we're concerned with the simulation of σ_j, for $1 \leqslant j \leqslant p_1$. Then, the simulated trajectory and τ_j are skeleton equivalent, and the simulating trajectory and $\tau_{\theta(j)}^*$ are skeleton equivalent.

PROOF OF THEOREM 4: Let $c \in CON$ be a finite configuration. We now define a partition of Z^n consisting of four sets X_c, Y_c, Z_c, W_c. Let $d_c \geqslant 0$ be such that for all $\underline{\zeta} \in Z^n$, if the absolute value of at least one coordinate of $\underline{\zeta}$ is greater than or equal to d_c, then $c(\underline{\zeta})$ is quiescent. For $\underline{\alpha} \in (a_1, \ldots, a_n) \in Z^n$, let $size(\underline{\zeta}) = \max(|a_1|, \ldots, |a_n|)$. We then let $X_c = \{\underline{\zeta} \mid \underline{\zeta} \in Z^n$ and $size(\underline{\zeta}) \leqslant d_c\}$. Suppose $NI = \langle \underline{\gamma}_1, \ldots, \underline{\gamma}_q \rangle$. Letting $e = \max(size(\underline{\gamma}_1), \ldots, size(\underline{\gamma}_q))$, we let $Y_c = \{\underline{\zeta} \mid \underline{\zeta} \in Z^n$ and $d_c < size(\underline{\zeta}) \leqslant d_c + e\}$, $Z_c = \{\underline{\zeta} \mid \underline{\zeta} \in Z^n$ and $d_c + e < size(\underline{\zeta}) \leqslant d_c + 2e\}$, and $W_c = \{\underline{\zeta} \mid \underline{\zeta} \in Z^n$ and $d_c + 2e < size(\underline{\zeta})\}$. For each $\tau \in \{\tau_1, \ldots, \tau_{p_1}, \tau_{\theta(1)}^*, \ldots, \tau_{\theta(p_1)}^*\}$ let R_τ be as in the proof of Theorem 3. Letting $S_{\tau,c} = CS_\tau(X_c \cup Y_c \cup Z_c)$, we have that $R_\tau \cap (\mathcal{P}(S_{\tau,c}) \times \mathcal{P}(S_{\tau,c}))$ is a partial order over $\mathcal{P}(S_{\tau,c})$. We embed this partial order in a total order $T_{\tau,c}$. Let $\{\psi_1, \ldots, \psi_\ell\}$ be a complete listing of $\{\tau(i) \mid i \in IS_\tau(X_c \cup Y_c \cup Z_c)\}$ such that, if $1 \leqslant r < s \leqslant |IS\ (X_c \cup Y_c \cup Z_c)|$ then $\tau(r)\ T_{\tau,c}\ \tau(s)$. We now define a trajectory τ_c' over Z^n with the same skeleton set as τ but which is such that $\tau_c'(0) = \psi_\ell$, $\tau_c'(1) = \psi_{\ell-1}, \ldots, \tau_c'(\ell-1) = \psi_1$. We will now show that for σ the local transformation associated with τ, the successor configuration of c processed with respect to σ and τ_c', which we call $\hat{c}$, is the same as the successor configuration of c processed with respect to σ and a trajectory for pure parallel processing, which we call $\bar{c}$. It is easily seen that $\bar{c}(\underline{\zeta})$ is quiescent for $\underline{\zeta} \in Z_c \cup W_c$. We now show that $\hat{c}(\underline{\zeta})$ is also quiescent for $\underline{\zeta} \in Z_c \cup W_c$. Suppose $\underline{\zeta} \in Z_c$. Let $\{\psi_{i_1}, \ldots, \psi_{i_k}\} \subseteq \{\psi_1, \ldots, \psi_\ell\}$ be a listing of $\{\tau(i) \mid i \in IS_\tau(Z_c)\}$, where $i_1 < \ldots < i_k$. The ordering of processing Z_c is thus to start with ψ_{i_k} and to proceed to ψ_{i_1}. Now, the ordering is such that ψ_j, $i_k < j \leqslant \ell$ cannot be a related set of ψ_{i_k}. Thus, if $\underline{\zeta} \in \psi_{i_k}$, when $\underline{\zeta}$ is processed, all its neighbors are quiescent and hence, it remains quiescent. The same argument can be made if $\underline{\zeta} \varepsilon \psi_{i_j}$ for $1 \leq j < k$. Thus, if $\underline{\zeta} \varepsilon Z_c$, we have $\hat{c}(\underline{\zeta})$ is quiescent. If $\underline{\zeta} \varepsilon W_c$, the neighbors of $\underline{\zeta}$ can only be in $Z_c \cup W_c$. An argument similar to the above can be made to show that $\hat{c}(\underline{\zeta})$ is again quiescent. Now, for $\underline{\zeta} \varepsilon X_c \cup Y_c$, whenever $\underline{\zeta}$ is processed, the neighbors of $\underline{\zeta}$ are in the same states as they were in configuration c, since clumps processed before the clump containing ζ are not related sets of this latter clump. Thus, for $\underline{\zeta} \varepsilon X_c \cup Y_c$, $\overline{c}(\underline{\zeta}) = \hat{c}(\underline{\zeta})$.

Thus, for a given $c \varepsilon$ CON, letting $\hat{\tau}_1 = (\tau_1)'_c, \ldots, \hat{\tau}_{p_1} = (\tau_{p_1})'_c$, $\overline{\tau}_{\theta(1)} = (\tau^*_{\theta(1)})'_c, \ldots, \overline{\tau}_{\theta(p_1)} = (\tau^*_{\theta(p_1)})'_c$, and for $j \notin \mathrm{range}(\theta)$, letting $\overline{\tau}_j$ be arbitrary, we have our result. (See notation for Definition 12). QED

THEOREM 5: Let $STA = <S, Z^n, NI, LT, <\tau_1, \ldots, \tau_p>>$ be a stratified tessellation automaton such that, for each $1 \leq i \leq p$, $|\{\mathrm{type}^{(STA)}_{\underline{\zeta},\tau_i} \mid \underline{\zeta} \varepsilon Z^n\}| < \aleph_0$. Suppose that $\overline{\tau}_1, \ldots, \overline{\tau}_p$ are any p trajectories over Z^n. Then, there is another stratified tessellation automaton $\overline{STA} = <\overline{S}, Z^n, \overline{NI}, \overline{LT}, <\overline{\tau}_1, \ldots, \overline{\tau}_p>>$ which simulates STA in 2 times real time in a proper manner. This simulation is such that Δ preserves finite configurations.

PROOF OF THEOREM 5: This theorem will follow from the proof of the following two lemmas:

LEMMA 1: Let $STA = <S, Z^n, NI, LT, <\tau_1, \ldots, \tau_p>>$ be a stratified tessellation automaton such that, for each $1 \leq i \leq p$, $|\{\mathrm{type}^{(STA)}_{\underline{\zeta},\tau_i} \mid \underline{\zeta} \varepsilon Z^n\}| < \aleph_0$. Suppose that $\tau'_1, \ldots, \tau'_p$ are any p trajectories over Z^n for pure parallel processing. Then, there is another stratified tessellation automaton $STA' = <S', Z^n, NI', LT', <\tau'_1, \ldots, \tau'_p>>$ which simulates STA in real time in a proper manner. This simulation is such that Δ preserves finite configurations.

PROOF OF LEMMA 1: We here illuminate Definition 13, that of type. In order to simulate an arbitrary mode of processing by a pure parallel mode, we need to know how each cell is processed sequentially; all the information concerning the ordering of processing of each cell must be available. The type of a cell is an indicator which provides this information; it gives the history of the computation which preceded the processing of the given cell. This history of computation, in turn, will indicate how many cells have been processed and in what order. This information is essential in processing any cell, for it is necessary to know the updated values of the neighbors of that cell and how these neighbors were processed. Thus, associated with the type of a cell is the information of how many and which cells must be taken into account in processing it. The scheme here is to alter and expand the original neighborhood structure in order to capture the total history of computation of those cells which may affect the processing of the cell in question. Since the types of the cells tells us how many and which cells are important, that information will be used in the construction of the expanded neighborhood.

We now explain the notion of the set of cells that affect the processing of a particular cell $\underline{\zeta}$ as indicated by the type of $\underline{\zeta}$. But first, let us introduce an ordering on the various levels of types. For $\alpha \in \bigcup_{i \geq 1} S_i$, let $d(\alpha) = \min\{j \mid \alpha \in S_j \text{ and } \alpha \notin S_{j-1}\}$. For $\alpha, \beta \in \bigcup_{i \geq 1} S_i$, define $\alpha \simeq \beta$ if $d(\alpha) = d(\beta)$. Then, $\simeq$ is an equivalence relation. Let $/\alpha/$ be the equivalence class of α with respect to $\simeq$. Let $/\alpha/ \leq^* /\beta/ \leftrightarrow d(\alpha) \leq d(\beta)$. So $\leq^*$ is a total order on $(\bigcup_{i \geq 1} S_i)/\simeq$. From this discussion, the following claim becomes immediate: Let STA = $\langle S, Z^n, NI, LT, TR \rangle$, $\underline{\zeta} \in Z^n$ and $\tau \in TR$. Suppose $\text{type}^{(STA)}_{\underline{\zeta},\tau} = \langle \beta_1, \ldots, \beta_r \rangle$. Then, for $1 \leq i \leq r$, if $\beta_i \neq *$, we have that $\beta_i = \text{type}^{(STA)}_{\underline{\xi},\tau}$ for some $\underline{\xi} \in Z^n$; in fact, $/\beta_i/ \leq^* /\text{type}^{(STA)}_{\underline{\zeta},\tau}/$.

Now, for $1 \leq i \leq p$, let $TYPE_{\tau_i} = \{\text{type}^{(STA)}_{\underline{\zeta},\tau_i} \mid \underline{\zeta} \in Z^n\} = \{\nu_1, \ldots, \nu_m\}$, where $/\nu_1/ <^* \ldots <^* /\nu_m/$. (Note that $\nu_1 = (*, \ldots, *) \in \{*\}^q$.) For $1 \leq k < q$, we now associate a set of points of Z^n with each ν_k, called Λ_k. This set will serve as a neighborhood index for all points of type ν_k. That is, to find the successor state of a cell $\underline{\zeta}$ of type ν_k, with respect to the trajectory τ_i, it suffices to consider only the present states of $\underline{\zeta} + \Lambda_k = \{\underline{\zeta} + \underline{\xi} \mid \underline{\xi} \in \Lambda_k\}$. Let $\Lambda_1 = \{\underline{\gamma}_1, \ldots, \underline{\gamma}_q\} = NI$. This is obviously so because $\nu_1 = (*, \ldots, *) \in \{*\}^q$. For $1 \leq k < m$, define Λ_{k+1} as follows:

Suppose $\nu_{k+1} = (\beta_1, \ldots, \beta_q)$. Thus, for $1 \leq i \leq q$, β_i is either * or some ν_j, $j < k+1$. Now, $\Lambda_{k+1} = \Sigma_1^{(k+1)} \cup \ldots \cup \Sigma_q^{(k+1)}$, where for $1 \leq i \leq q$,

$$\Sigma_i^{(k+1)} = \begin{cases} \underline{\gamma}_i & \text{if } \beta_i = * \\ \underline{\gamma}_i + \nu_j & \text{if } \beta_i = \nu_j,\ j < k+1 \end{cases}$$

Let $N_{\tau_i} = \Lambda_1 \cup \ldots \cup \Lambda_m$, and let $N' = \bigcup_{1 \leq i \leq p} N_{\tau_i} \cup \{\underline{0}\}$. Now, let $I = (\underline{u}_1, \ldots, \underline{u}_d)$ be an arbitrary ordering of N'. We will then have that $S' = S \times \mathop{\times}\limits_{1 \leq i \leq p} TYPE_{\tau_i}$, $NI' = I$. Since space does not permit, we do not specify LT'. For a more thorough explication and an example, one may consult Tsui [12], page 74.

LEMMA 2: Let STA = $\langle S, Z^n, NI, LT, \langle \tau_1, \ldots, \tau_p \rangle \rangle$ be a stratified tessellation automaton such that $\tau_1, \ldots, \tau_p$ are trajectories for pure parallel processing. Suppose that $\tau'_1, \ldots, \tau'_p$ are any p trajectories over Z^n. Then, there is another stratified tessellation automaton STA' = $\langle S', Z^n, NI', LT', \langle \tau'_1, \ldots, \tau'_p \rangle \rangle$ which simulates STA in two times real time in a proper manner. This simulation is such that Δ preserves finite configurations.

PROOF OF LEMMA 2: This is similar to the proof of Theorem 2 in Grosky and Tsui [5]. QED

This completes our discussion of Theorem 5.

DEFINITION 14: Suppose STA' = $\langle S', Z^n, NI', LT', TR' \rangle$ simulates STA = $\langle S, Z^n, NI, LT, TR \rangle$. We say that Δ has the local property if there is a function $f: S' \to S$ such that if $c \in CON$ and $\underline{\zeta} \in Z^n$, then $f(\Delta(c)(\underline{\zeta})) = c(\underline{\zeta})$.

DEFINITION 15: Let STA = $\langle S, Z^n, NI.LT, TR \rangle$ be a stratified tessellation automaton. If $\sigma \in LT$, we say that σ has the expanding property if, whenever

$s \in S$ is quiescent with respect to σ, then $\sigma(s_1,\ldots,s_q) = s \rightarrow s_1 = s_2 = \ldots = s$.

We present the following theorem without proof:

THEOREM 6: Suppose $STA' = \langle S', Z^n, NI', \langle\sigma'_1,\ldots,\sigma'_{p_2}\rangle, \langle\tau'_1,\ldots,\tau'_{p_2}\rangle\rangle$ simulates $STA = \langle S, Z^n, NI, \langle\sigma_1,\ldots,\sigma_{p_1}\rangle, \langle\tau_1,\ldots,\tau_{p_1}\rangle\rangle$ in real time in a proper manner, such that Δ has the local property. Assume that $\sigma_1,\ldots,\sigma_{p_1}$ have the expanding property. Let $A = \{i \mid 1 \leq i \leq p_2$ and there exists $1 \leq k \leq p_1$ such that $\Gamma(\sigma_k) = \sigma'_i\}$. For $i \in A$, suppose that $z_i = |\{\text{type}^{(STA')}_{\underline{\zeta},\tau'_i} \mid \underline{\zeta} \in Z^n\}| < \aleph_0$. Then, for $1 \leq j \leq p_1$, $y_i = |\{\text{type}^{(STA)}_{\underline{\zeta},\tau_j} \mid \underline{\zeta} \in Z^n\}| < \aleph_0$.

REFERENCES

(1) S. Amoroso and G. Cooper, 'Tessellation Structures for Reproduction of Arbitrary Patterns,' *Journal of Computer and System Sciences*, v. 5 (1973), pp. 455-464

(2) S.K. Chang, 'On the Parallel Computation of Local Operations,' *Proc. of the 3rd Annual ACM Symposium on the Theory of Computation*, Shaker Heights, Ohio (1971), pp. 101-115

(3) W.I. Grosky and F. Tsui, 'Evolution of Patterns in Cellular Structures with Non-Uniform Parallel Transformations', *Proc. of the IEEE Systems, Man & Cybernetics*, Boston, Massachusetts (1973) pp.169-173

(4) ———, 'Parallel-Sequential Processing of Finite Patterns,' *Proc. of the 2nd Sagamore Computer Conf. on Parallel Processing*, Syracuse University, Syracuse, New York (1973), pp. 61-68

(5) ———, 'Pattern Generation in Non-Standard Tessellation Automata,' *Proc. of the ACM National Conf.*, Atlanta, Ga. (1973), pp. 345-348

(6) E. Lieblein, *A Theory of Patterns in Two-Dimensional Tessellation Space*, Ph.D. Thesis, Dept. of E.E., Univ. of Pa., 1968

(7) M. Minsky and S. Papert, *Perceptrons*, MIT Press, Cambridge, Mass.1969

(8) A. Rosenfeld, 'Isotonic Grammars, Parallel Grammars, and Picture Grammars,' *Machine Intelligence 6*, (eds.) B. Meltzer and D. Michie, Edinburgh Univ. Press (1970), pp. 281-294

(9) ——— and D.L. Milgram, 'Array Automata and Array Grammars,' *IFIP Congress 71*, Amsterdam, North Holland Pub. Co. (1971), pp. 166-173

(10) ———, 'Parallel/Sequential Array Automata,' *Information Processing Letters*, v. 2 (1973), pp. 43-45

(11) A. Rosenfeld and J.L. Pfaltz, 'Sequential Operations in Digital Picture Processing,' *Journal of the ACM*, v. 13 (1966), pp. 471-494

(12) F. Tsui, *Parallel and Sequential Operations with Array Processors*, Ph.D. Thesis, School of I.C.S., Ga. Tech., 1974

(13) H. Yamada and S. Amoroso, 'A Completeness Problem for Pattern Generation in Tessellation Automata,' *Journal of Computer and System Sciences*, v. 4 (1970), pp. 137-176

STOCHASTIC TREE SYSTEMS FOR SYNTACTIC PATTERN RECOGNITION[1]

B. K. Bhargava[2] and K. S. Fu
School of Electrical Engineering
Purdue University
West Lafayette, Indiana 47907

I. INTRODUCTION

Tree systems have been applied to syntactic pattern recognition [1,2, 3,4]. It has been found that when considering practical applications, a certain amount of uncertainty exists in the process under study. In order to describe and recognize noisy samples under possible ambiguous situations, the use of stochastic languages has been suggested [9]. By associating probabilities with the coded structures of samples, a probability measure can be imposed on the language describing the sample class. Thus statistical decision rules can be applied to the classification of patterns under ambiguous situations. Though this area of syntactic pattern recognition is yet to be explored in depth, stochastic grammars and stochastic automata have been studied for strings in the last few years [5]. There is hardly any formalism available for stochastic multi-dimensional grammars, except for probabilistic tree automata [6,8].

In this paper, we present a formalism for stochastic tree system and the application of this formalism to the classification of bubble chamber photographs. The stochastic tree grammars and stochastic tree automata are defined and the relationship among them is studied. Next the properties of stochastic tree languages are presented and it has been found that the consistency criterion and average tree depth are similar to those obtained for context-free grammars. More definitions can be found in [2,10].

In a hydrogen bubble chamber, if the incoming particles are positively charged, the photographs with odd number of prongs are distortions of two, four or six prong photographs. In a deterministic tree system [1,2,3], such photographs are rejected, but using the probability distribution and employing the stochastic tree system, such photographs can be classified in one of the possible classes.

II. BASIC DEFINITIONS

Definition 1: A stochastic tree grammar (S.T.G.) G_S is a 4-tuple $G_S = (V,r',P,S)$ over ranked alphabet $\langle V_T,r \rangle$ where

$\langle V,r' \rangle$ is a finite ranked alphabet such that $V_T \subseteq V$ and $r'/V_T = r$. The elements of V_T and $V - V_T = V_N$ are called terminals and nonterminal symbols respectively.
P.is a finite set of stochastic production rules of the form $\phi \overset{P}{\rightarrow} \Psi$ where ϕ and Ψ are trees over $\langle V,r' \rangle$ and $0 \leq P \leq 1$.
$S \subseteq T_V$ is a finite set of axioms.

Definition 2: $\alpha \xrightarrow{a,p} \beta$ is in G_S if and only if there is a stochastic production $\phi \xrightarrow{P} \Psi$ in P such that $\alpha/a = \phi$ and $\beta = (a \leftarrow \Psi)\alpha$; i.e., ϕ is a subtree of α at "a" and β is obtained by replacing the occurrence of ϕ at "a" by ψ. We write $\alpha \overset{P}{\rightarrow} \beta$ in G_S if and only if there exists $a \in D\alpha$ such that $\alpha \xrightarrow{a,P} \beta$.

---

[1] This work was supported by the National Science Foundation Grant GK-36721.

[2] Dr. Bhargava is currently with the Regenstrief Institute, Indianapolis, Indiana.

Definition 3: If there exists a sequence of trees $t_0, t_1, \dots, t_m$ such that

$$\alpha = t_0,\ \beta = t_m,\ t_{i-1} \Rightarrow t_i,\ i = 1,\dots,m$$

then we say that α generates β with probability $p = \prod_{i=1}^{i=m} p_i$ and denote this derivation by $\alpha \overset{p}{\vdash} \beta$ or $\alpha \overset{p}{\Rightarrow} \beta$.

The sequence of trees $t_0, \dots, t_m$ is called a derivation of β from α. The probability associated with this derivation is equal to the product of the probabilities associated with the sequence of stochastic productions used in the derivation.

Definition 4: The tree language generated by a S.T.G. G_S is

$$L(G_S) = \{t, p(t) \mid t \in T_{V_T},\ S \overset{p_i}{\Longrightarrow} t,\ i = 1,\dots,k \text{ and } p(t) = \sum_{i=1}^{k} p_i\}$$

where k is the number of all distinctly different derivation of t from S and p_i is the probability associated with the i-th distinct derivation of t from S.

A stochastic tree grammar $G_s = (V,rP,S)$ over $\langle V_T, r\rangle$ is simple if and only if all rules of P are of the form

$$X_0 \xrightarrow{p} \begin{array}{c} x \\ \diagup \quad \diagdown \\ X_1,\dots,X_{r(x)} \end{array},\quad X_0 \xrightarrow{q} X_1,\ \text{or}\quad \begin{array}{c} x \\ \diagup \quad \diagdown \\ X_1,\dots,X_{r(x)} \end{array} \xrightarrow{r} X_0$$

where $X_0, X_1, \dots, X_{r(x)}$ are nonterminal symbols and $x \in V_T$ is a terminal symbol and $0 < p,q,r \leq 1$.

Lemma. Given a stochastic tree grammar $G_s = (V,r,P,S)$ over V_T, one can effectively construct a simple stochastic tree grammar $G_s' = (V',r',P'S')$ over V_T which is equivalent to G_s.

Definition 5: Let $\langle \Sigma, r\rangle$ be a ranked alphabet, where $\Sigma = (x_1, \dots, x_R)$. A stochastic climbing tree automaton over Σ (or V_T for tree grammar) is a system

$$M_s = \langle Q, f_1, f_2, \dots, f_k, q_0, F\rangle$$

where

Q is a finite set of states
$q_0 \in Q$ is the initial state
$F \in Q$ is the set of final states
For each x_i, $1 \leq i \leq K$,
f_i is a relation on $Q \text{ X } [0,1] \text{ X } Q^{r(x_i)}$
the interpretation being that a state q under an input x_i enters

a string of state $q_1 \text{ X} q_2 \dots q_{r(x_i)}$ $\quad Q^{r(x_i)}$ with a transition

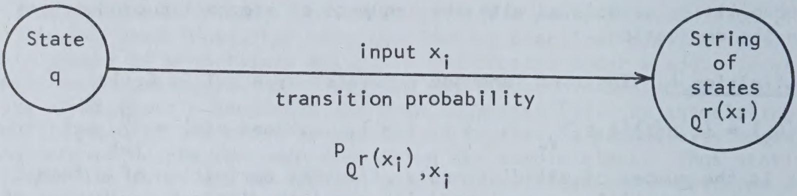
probability $p_{Q^{r(x_i)}}$.

$$\text{e.g., } f_i\colon q \text{ X } x_i \xrightarrow{p_{Q^{r(x_i)}},\, x_i} (q_1 \text{ X } q_2 \dots \text{X } q_{r(x_i)})$$

where $q_1, q_2, \dots, q_{r(x_i)},\ q \in Q$

and $Q^{r(x_i)}$ is a string of states $q_1 X q_2 \dots X q_{r(x_i)}$. Also for all states $q \varepsilon Q$, the total probability of leaving q is 1 (i.e., $\sum\limits_{x \varepsilon \Sigma} p_{q,x} = 1$), $0 < p_{q,x} \leq 1$, where $p_{q,x}$ is the probability of transition from state q to all strings of states $Q^{r(x)}$, with an input x.

If the above condition is not satisfied, the above will be called weighted climbing tree automaton.



Notation: If $x = x_i \varepsilon \Sigma$, then f_x means f_i.

Definition 6: The response relation ρ of a stochastic tree automaton M_S is defined as follows:

(a) if $x \varepsilon \Sigma_0$, $\rho(x) \to (q, p_{q,x})$

if and only if $f_x \xrightarrow{p_{q,x}} q$

(b) if $x \varepsilon \Sigma_n$, $n > 0$, $\rho\left(\overset{x}{\diagup\!\diagdown}_{t_1 \dots t_n}\right) \to \left(Q^{r(x)}, p_{Q^{r(x)},x}\right)$

if and only if there exists $q, q_1, q_2, \dots, q_n \ \varepsilon\ Q$,

$$f_x(q) \xrightarrow{p_{Q^{r(x)},x}} Q^{r(x)}$$

and $\rho(t_i) \to (q_i, p_{q_i,t_i})$

and $Q^{r(x)} = q_1 X q_2 X \dots q_n$.

Note that $t_i \ \varepsilon\ T$ for $1 \leq i \leq n$. Also, ρ is a function ρ: T ρ: $T_\Sigma \to (Q^s, p(t))$.

Note Q^s is a string of states. If $t = \overset{x}{\diagup\!\diagdown}_{t_1 \dots t_n}$,

$$p(t) = p_{Q^{r(x)},x} \left(\prod_{i=1}^{n} p_{q_i,t_i} \right)$$

Definition 7: The language accepted by M_S with a cut-point λ such that $0 \leq \lambda \leq 1$ is

$$L(M_s,\lambda) = \{t \ \varepsilon\ T_\Sigma \mid Q^s \ \varepsilon\ F \text{ such that } \rho(t) \xrightarrow{p(t)} Q^s \text{ and } p(t) > \lambda\}$$

$L(M_S,\lambda)$ is called the λ-stochastic tree language.

We do not know if λ-stochastic tree language is more powerful than regular set.

Definition 8: The cut-point of a stochastic climbing tree automaton M_s is called isolated if it is an isolated point of set $\{p(t)/t \varepsilon t_\Sigma\} \cup \{\lambda\}$. (We can assume $\lambda \neq p(t)$ for all t - otherwise we move "λ" a little.

Theorem 1 [6]: For any stochastic climbing automaton M_s with an isolated cutpoint there exists a climbing automaton M_t such that $L(M_s) = L(M_t)$.

Proof. The proof for the above theorem can be found in [6].

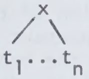
Definition 9: Two stochastic tree automata M_{s_1} and M_{s_2} are equivalent if and only if $L(M_{s_1}) = L(M_{s_2})$.

Definition 10: A stochastic tree grammar G_s is equivalent to a stochastic tree automaton M_s if and only if $L(G_s) = L(M_s)$.

Theorem 2 [2]: For every normalized stochastic expansive tree grammar $G_s = (V,r,P,S)$ over V_T, one can effectively construct a normalized stochastic climbing tree automaton M_s such that $L(M_s) = L(G_s)$ and vice versa.

It is important to note that there is exactly one to one correspondence between the productions of G_s and transitions of M_s and the probability of production and transition is exactly the same. Further, for a tree $t \varepsilon L(G_s)$, the set of production rules used for generation exactly correspond one to one with the set of transition functions used for acceptance of the same tree $t \varepsilon L(M_s)$.

Hence, the probability of generation p(t) of a tree $t \varepsilon L(G_s)$ in G_s is exactly the probability of acceptance p(t) of tree $t \varepsilon L(M_s)$ accepted by M_s

and $p(t) = p_{X^n,x} \times \prod_{i=1}^{n} p(t_i)$, where $t =$ 

and $p(t) = p_{X,x}$, where $t = x$.

III. APPLICATION TO SYNTACTIC PATTERN RECOGNITION

In [1], we have noticed that a deterministic tree system completely rejects the three and five prong photographs since they are not generated by the tree grammar formulated for events. Such photographs represent interactions in which either some secondary tracks are too small or overlap around the vertex and can not be detected as separate tracks by the computer. Since we obtained the data for photographs which represent only events, a statistical distribution of topology for given photographs is preknown and is as shown in Table 1. It is interesting to note that photographs with charge of two positive and one negative on secondary particles are found to be in four prong class more than two prong and six prong class. Similarly, photographs with charge of three positive and two negative occur more often in the six prong class of events.

In this section we use the formalism for stochastic tree systems obtained in previous sections to classify three and five prong events. The decision making process for classification uses apriori probabilities and maximum likelihood criterion given in [5] and is briefly described as follows:

Let $C = \{G_{s_i}, i = 1,\ldots,N\}$ be a finite set of stochastic tree grammars, where G_{s_i} generates a set of trees $L(G_{s_i})$ which are primitive structural descriptions of possible observations of pattern defined in a pattern class

ω_i in a random or noisy environment and defines a probability distribution over the set of trees $L(G_{s_i})$. $P(t|G_{s_i},C)$ is defined as the probability of the tree t being generated by a stochastic tree grammar G_{s_i} in C and $P(G_{s_i}|t,C)$ as the a posteriori conditional probability of G_{s_i}, given the tree t and G_{s_i} in C. Furthermore, it is assumed that each grammar has a known a priori probability of being correct, i.e. $P(G_{s_i}|C)$ is known and

$$\sum_{G_{s_i} \varepsilon C} P(G_{s_i}|C) = 1.$$

The decision rule will choose the pattern class ω_i as the pattern class which contains the unknown pattern represented by t if

$$P(G_{s_i}|t,C) \geq P(G_{s_j}|t,C), \text{ for all } j \neq i.$$

Since

$$P(G_{s_i}|t,C) = P(G_{s_i}|C)P(t|G_{s_i},C)/P(t|C)$$

where

$$P(t|C) = \sum_j P(t|G_{s_i},C)$$

is independent of i, to maximize the a posteriori conditional probability $P(G_{s_i}|t,C)$ is equivalent to maximize

$$P'(G_{s_i}|t,C) = P(G_{s_i}|C)P(t|G_{s_i},C)$$

Usually, we are given a set of samples (trees) belonging to a class and the statistical distribution of their occurrence. To construct a stochastic tree grammar to represent these samples, we need to infer the production probabilities from the occurrence distribution. Since tree languages can be modelled by a generalized branching process in the same manner as context-free languages [7], the theory of statistical inference of Markov processes can be used to infer the set of production probabilities D where $D = \{\hat{p}_{ij} | A_i \rightarrow \phi_j \ \varepsilon \ P \text{ and } A_i \ \varepsilon \ V_N, \ \phi_j \ \varepsilon \ T_V\}$

Thus
$$\hat{p}_{ij} = \frac{n_{ij}}{\sum_j n_{ij}} \qquad (1)$$

where $\hat{p}_{ij}$ is the maximum likelihood estimate for p_{ij} and n_{ij} follows a multi-normal distribution with probability p_{ij}.

Further,

$$n_{ij} = \sum_{t \ \varepsilon \ S_{V_T}} n_t N_{ij}(t) \qquad (2)$$

where S_{V_T} is the given set of sample trees, n_{ij} is the number of times production $A_i \rightarrow \phi_j$ is used in the generation of all trees in the estimate. n_t is the number of times tree t appears in the estimate. $N_{ij}(t)$ is the number of times production $A_i \rightarrow \phi_j$ appears in the generation of t. $\sum_j$ is the sum over all the right parts ϕ_j for a given A_i as left part.

Substituting equation (2) into (1), we get

$$\hat{p}_{ij} = \frac{\displaystyle\sum_{t \in S_{V_T}} n_t N_{ij}(t)}{\displaystyle\sum_{j} \sum_{t \in S_{V_T}} n_t N_{ij}(t)} \tag{3}$$

For the supervised single class production probability inference procedure, the inferred probability $\hat{p}_{ij}$ defined by (1) approaches p_{ij}, the actual probability. Now we apply the above described machanism to classify the otherwise rejected photographs in a bubble chamber experiment.

Let G_{S_2}, G_{S_4}, and G_{S_6} be the stochastic tree grammars representing the classes of two, four, and six prong events respectively. The probabilities on the productions are obtained using Table 1 and the procedure described above. Each production with symbol S on the left-hand side of the production rule decides the type of photograph generated.

Thus we have

$G_{S_2} = (V,r,P,S)$

$V = \{S,A,B,V_T\}$

$V_T = C_p$: track with convex curvature

C_n: track with concave curvature

$r(C_p) = \{3,2,0\}$

$r(C_n) = \{0\}$

$$P: \quad S \xrightarrow{0.95} \begin{array}{ccc} & C_p & \\ / & & \backslash \\ A & & A \end{array}$$

$$S \xrightarrow{0.03} \begin{array}{ccc} & C_p & \\ / & | & \backslash \\ A & A & B \end{array}$$

$$S \xrightarrow{0.02} \begin{array}{ccc} & C_p & \\ / & & \backslash \\ A & & B \end{array}$$

$$A \xrightarrow{1.0} C_p$$

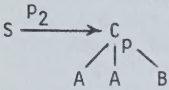
$$B \xrightarrow{1.0} C_n$$

The production probability p_1 in production $S \xrightarrow{p_1} \begin{array}{ccc} & C_p & \\ / & & \backslash \\ A & & A \end{array}$ is obtained by using the procedure described earlier and row one in Table 1.

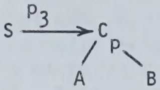
$$\text{i.e.} \quad p_1 = \frac{95 \times 1 + 3 \times 0 + 2 \times 0}{95 \times 1 + 3 \times 1 + 2 \times 1}$$

$$= \frac{95}{100} = 0.95$$

Similarly, production probabilities p_2 and p_3 associated with productions



and



are given as

$$p_2 = \frac{95\text{x}0 + 3\text{x}1 + 2\text{x}0}{95\text{x}1 + 3\text{x}1 + 2\text{x}1} = .03$$

and

$$p_3 = \frac{95\text{x}0 + 3\text{x}0 + 2\text{x}1}{95\text{x}1 + 3\text{x}1 + 2\text{x}1} = .02$$

The productions $A \rightarrow C_p$ and $B \rightarrow C_n$ are used in every sample and since have no effect on the occurrence distribution, have probability 1.0 associated with them. The contents of Table 1 are obtained by studying each class of events separately and noting the charges on the secondary particles as interpreted by the computer system. Using the second row and the third row in Table 1, and following the above steps, we obtain the production probabilities of stochastic tree grammars G_{S_4} and G_{S_6}.

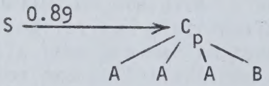
$G_{S_4} = (V,r,P,S)$

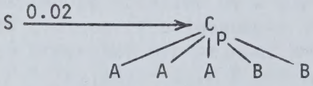
where V and V_T are as before.

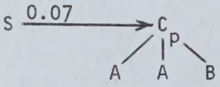
r: $r(C_p) = \{5,4,3,0\}$

$r(C_n) = \{0\}$

P:







$A \xrightarrow{1.0} C_p$

$B \xrightarrow{1.0} C_n$

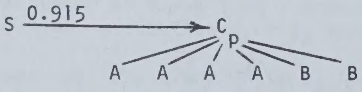
and $G_{S_6} = (V,r,P,S)$

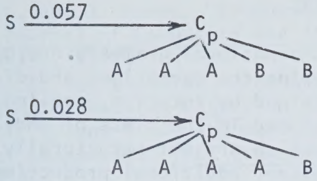
V and V_T as before

$r(C_p) = \{6,5,0\}$

$r(C_n) = \{0\}$

P:





$$A \xrightarrow{1.0} C_p$$

$$B \xrightarrow{1.0} C_n$$

Now, we have

$$C = \{G_{t_2}, G_{t_4}, G_{t_6}\}$$

$$P(G_{t_2}|C) = \frac{100}{306}$$

$$P(G_{t_4}|C) = \frac{100}{306}$$

and $P(G_{t_6}|C) = \frac{106}{306}$

For a tree t representing a three prong event with two positively and one negatively charged secondary tracks, we get

$$P(t\ G_{t_2},C) = \frac{3}{100}$$

and $P(t\ G_{t_4},C) = \frac{7}{100}$

Thus, $P'(G_{t_2|t},C) = P(G_{t_2}|C)P(t|G_{t_2},C)$

$$= \frac{100}{306} \times \frac{7}{100}$$

$$= \frac{7}{306}$$

Since $P'(G_{t_4}\ t,C) > P'(G_{t_2}\ t,C)$, we classify this photograph in a four prog class.

This decision making procedure was implemented on a CDC 6500 to classify the bubble chamber photographs. It is pointed out that though a decision was made on each photograph to be included in one or the other class, several photographs previously correctly rejected were misclassified. The misclassification rate was significantly smaller than rejection rate and a satisfactory classification system was achieved. In all, five photographs were misclassified as compared to twenty-five photographs rejected previously.

The results of classification using stochastic tree system are presented in Table 2. It is pointed out that tree transformations on photographs as described in [4] were used along with probability information.

As pointed out earlier, the misclassified events were forced in one of the classes using probability information.

IV. REMARKS

We have found in [4] that tree transformational grammars are useful in syntactic pattern recognition for considering the variations and distortions among patterns. Transformed patterns obtained by rotation, scaling, deletion and permutation, etc. are easily included in the class of undistorted patterns. To consider a set of patterns which are not structurally different so as to be separated in distinct classes, additional productions in the generative grammar become necessary. If the variation occurs periodically or in a rather well defined manner, it is convenient to include, a set of transformation rules rather than add complex production rules for every small allowable variation, in the generative and recognition process.

Sometimes variations occur in a random manner, so it will be inefficient to include a transformation rule for every possible distortion. Further some distorted patterns may belong to several completely different classes. In such cases if the probability of a pattern occuring in one class is more than of its belonging to other classes, a probabilistic recognition system can be employed. The ambiguity of classifying a given pattern in one class in a multiclass recognizer can be eliminated by obtaining the probability of its occurrence in different classes and using maximum likelihood decision procedures.

Thus, transformational grammars are useful when allowing variations in a set of samples in a single class and stochastic systems are needed to avoid ambiguity in a multiclass classifier. In real situations, since ambiguities taken care of by probabilistic models occur less often as compared to the large number of variations in a given set of patterns, the transformational grammars become an integreted part of a pattern recognition system. Thus, if the classification results after using transformational grammars are found to be unsatisfactory, stochastic grammars should be employed next. Of course, a combination of both transformational grammars and stochastic grammars is usually desirable.

For the computer implementation, the use of transformational grammars requires storing the set of T-rules and the procedure of applying them. On the other hand, for stochastic grammars, the probabilities are attached to the production rules which thus becomes a part of the generation or recognition steps. Thus, a stochastic system is easier to implement as compared to a transformational grammar.

For classification of bubble-chamber photographs, we employed both techniques. Variations in photographs to consider changes in curvature of tracks and permutations among secondary tracks were easily taken care of by using tree-transformation rules. On the other hand, the ambiguity of deciding, three or five prong events as belonging to the two-prong, four-prong, or six-prong class, was eliminated by formulating stochastic tree grammars. Such ambiguity occurred for about twenty-five photographs while the variations accounted by transformations were present in more than two hundred photographs.

REFERENCES

1. Fu, K. S. and B. K. Bhargava, "Tree Systems for Syntactic Pattern Recognition," IEEE Trans. on Computers, Vol. C-22, pp. 1087-1099, Dec. 1973.
2. Bhargava, B. K., "Tree Systems for Syntactic Pattern Recognition," Ph.D. Thesis, Purdue University, W. Lafayette, Ind., March 1974.
3. Bhargava, B. K. and K. S. Fu, "Application of Tree System Approach to Classification of Bubble Chamber Photographs," TR-EE 72-30, Purdue Univ., Lafayette, Ind., Nov. 1972.
4. Bhargava, B. K. and K. S. Fu, "Transformation and Inference of Tree Grammars and Their Applications to Syntactic Pattern Recognition", IEEE SMC Conference, Oct. 2-4, 1974, Dallas, Texas.

5. Fu, K. S. and T. Huang, "Stochastic Grammars and Languages," International Journal of Computers and Information Sciences, Vol. 1, No. 2, pp. 135-170, 1972.
6. Magidor, M. and G. Moran, "Probabilistic Tree Automata and Context-Free Languages," Israel J. Math., Vol. 8, pp. 340-348, 1970.
7. Booth, T. L. and R. A. Thompson, "Applying Probability Measures to Abstract Languages," IEEE Transactions on Computers, Vol. C-22, pp. 442-450, 1973.
8. Ellis, C. A., "Probabilistic Languages and Automata," Information and Control, Vol. 19, pp. 401-416, 1971.
9. Fu, K. S., "Syntactic Pattern Recognition and Stochastic Languages," Frontiers of Pattern Recognition, ed. by S. Watanabe, Academic Press, 1972.
10. Brainerd, W. S., "Tree Generating Regular Systems," Information and Control, Vol. 14, pp. 217-231, 1969.

Table 1. Statistical Distribution of Topology of Photographs

Class of Events	Total No. of Events	No. of Events with Charge on Secondary Particles as								
		2+ 0-	2+ 1-	1+ 1-	3+ 1-	3+ 2-	3+ 0-	4+ 2-	4+ 1-	All Others
2	100	95	3	2	0	0	0	0	0	0
4	100	0	7	0	89	2	2	0	0	0
6	106	0	0	0	0	6	0	97	3	0
2,4,6	306	95	10	2	89	8	2	97	3	0

Note: + means positive charge
- means negative charge

Table 2. Classification of Photographs in Two, Four and Six Prong Classes Using Stochastic Tree Grammars

Type of Event	Total No. or Events	No. of Events Classified Correctly	No. of Events Classified Incorrectly
2	100	97	3
4	100	98	2
6	106	106	0

OPTIMAL FEATURES SELECTION ALGORITHM FOR A BINARY-PARTITIONED CLASSIFICATION SCHEME-ANALYSIS*

PAUL P. WANG, Associate Professor** and RICHARD C. BURNS, Graduate Student, Department of Electrical Engineering, Duke University and Department of Electrical and Computer Engineering, University of Mass.

ABSTRACT

This paper identifies the problem of selecting P features from M available features, where P ≤ M, for a complete classification or unique encoding for N patterns or classes. One may designate this problem as an (M,N) problem. Each feature is assumed to be binary-partitioned even though the algorithm can be generalized further. Theoretical analysis is carried out from the poset and dual gradic mapping view point. Two examples are being used as the vehicle of illustration throughout the paper. The dynamic programming technique may be used to obtain the optimum features selection from the "partitioning and decision tree," and the difficulties involved when the number N is large are also pointed out. The development presented in this paper is to "set the stage" for a more efficient heuristic programming technique in order to tackle a more complicated and practical pattern recognition problem.

I. Introduction

Features extraction has always been an important and active research area.[1]-[3] For most of the pattern classification problems, it is possible to extract a large number of features, however, not all the features are equally important. Hence the question of the ordering of these features must be raised and the answer to the importance of the ordering is very sensitive to the methodology and tools used to accomplish the pattern classification task. In this paper, two important assumptions are made that all features are binary coded into "1" or "0", and the scheme used to select the features is based upon the concept of "orthogonality" because it normally yields the minimum number of features necessary to classify a set of patterns successfully. The first assumption does not effect or lose any of the generality of the theory presented in this paper, that is the theory for multiple valued features can be developed in parallel with the binary coded case. As to the second assumption, the results of this paper are motivated by character recognition and the machine translation areas,[4] but the results should be equally applicable to the other problems particularly when the number of patterns is very large and the number of usable features is also larger than required to uniquely classify the patterns. Generally speaking, one may refer to this problem as a (M,N) problem, where the positive integer N represents the number of the patterns to be classified and the positive number M represents the total number of available features. In the final algorithm to be discussed later, the input to the algorithm is simply a binary-valued MxN matrix and the output is a submatrix PxN over the binary field where the positive integer P≤M, in general. In either of the matrices, input matrix and output matrix, the i^{th} row represents a row vector over the binary field as the result of choosing the feature f_i. In other words, a threshold (a real number on the whole real number line), which can be decided in such a manner that some patterns will be encoded as a "1" and the rest of the patterns being encoded as a "0".

---

* This research has been in part supported by a NASA research grant, NSG 5020 and in part by a Duke Research Council Regular Grant.

**On sabbatical leave from Duke University at University of Massachusetts.

Previous related research includes Henrichon and Fu's algorithm[5] which determines partitions that are parallel hyperplanes orthogonal to the feature coordinate dimensions and can be implemented in the form of a layered machine which is composed of multithreshold elements, and Meisel and Michalopoulos' algorithm[6] deals with efficient partitioning of a finite-dimensional space by the use of decision trees. The algorithm presented in this paper departs from the two algorithms mentioned above in a significant way, both in the definition of the problem (e.g., no transgenerator units are allowed as in [5]) as well as in the solution. Also in some sense, the algorithm may be used as an instrument for an information retrieval system but it departs from conventional thinking.[7]

In this paper, the mathematical model for the sequential selection of features as a binary partitioned decision tree is established. The basic properties associated with the decision trees are summarized. The rest of the paper is devoted to the solution of the problem through some mathematical derivations and proofs which constitute the essential facts and bases for the development of two algorithms to be presented in a forthcoming paper.

II. The Mathematical Model - Partition of a Number

By choosing an arbitrary feature f_i to partition a given set of N patterns into $(N, N-N_1)$ patterns, where N_1 of them are being coded as "1" and the rest $(N-N_1)$ are coded as "0", the feature f_i provides some information about the classification of the patterns and then when a second feature f_j has been chosen, it may partition the N patterns differently both in the numbers as well as the membership belonging to each class. To facilitate the analysis, it is helpful to illustrate the situation through a simple example. Assuming we have in hand the following a priori information of an MxN matrix; where N=6 and M=4, then patterns 3 and 4, $(N_1=2)$, belong to one class where all the rest, $(N-N_1=4)$, belong to the other class if f_1 is chosen.

$$\begin{array}{cc|cccccc|c}
 & & \multicolumn{6}{c}{\xrightarrow{N}} & \\
 & & 0&0&1&1&0&0 & f_1\\
M & \downarrow & 1&1&1&1&0&1 & f_2\\
 & & 0&0&1&0&0&1 & f_3\\
 & & 1&0&1&0&1&0 & f_4\\
 & & 1&2&3&4&5&6 &
\end{array}$$

On the other hand, if f_2 is chosen, then the partition would be (5,1). It is interesting to note that the MxN matrix is the input to the algorithm. So far as the pattern classification is concerned, one may try each possible combination of features sequentially. Hence, there are 4! or 24 possibilities even for this very simple problem.

Now we shall demonstrate the above mentioned problem is a "partition of a number" problem. In the sequel we denote the cardinality of a set A by m(A). A partition $\overline{n}$ on a positive number N is a set of positive integers whose sum is N and in which the same numbers are permitted as elements. Then one is able to write a partition on N, $\overline{n} = [N_1, N_2, \ldots, N_s]$ where $\sum_{i=1}^{s} N_i = N$. More generally, one may denote a partition $\overline{n}$ on N by $[(1)^{\lambda_1}(2)^{\lambda_2}\ldots(k)^{\lambda_k}\ldots(N)^{\lambda_N}]$ where $m(\{N_i \text{ in } \overline{n} \mid N_i = k\}) = \lambda_k$ for $1 \le k \le N$. This notation is called the standard form of a partition $\overline{n}$.[8] The set of all partitions of a positive integer N, (denoted by P(N)), a partial order is defined and a representative map of partitions is introduced. One can then define a dual gradic map of $(P(N),\le)$ and Narushima has shown that $(P(N),\le)$ is a graded poset.[8] We shall not expound further here concerning the definitions, propositions, theorems of dual gradic maps, graded posets. These can be found in reference [8]. Rather, we present in Fig. 1 a representative map of partitions and a dual gradic map of P(N), where N=6. Note that under the ordering for P(N) defined by Narushima,[8] the greatest element in P(N) is [(N)] and the least element is $[(1)^N]$. For $\overline{x}$ and $\overline{y}$ in P(N), Narushima designates $\overline{x\le y}$ if and only if $\overline{y}$ can be obtained from $\overline{x}$ after possible rearrangement by adding suitable parts, i.e., $\overline{x}=[x_{ij}|1\le i\le m, 1\le j\le r(i)]$, $\overline{y}=[y_i|1\le i\le m]$ and $y_i=\sum_{j=1}^{r(i)} x_{ij}$ for $1\le i\le m$. In the sequel, it will become clear that the solution to the pattern classification problem, which is the subject of our investigation, is to trace a path starting from the greatest element to the least element where $[(1)^N]$ is the answer we seek. In other words, the process of classification is considered "completed" if and only if one reaches the least element $[(1)^N]$.

To aid the understanding of the partitioning, we present in Fig. 1(b) a "partitioning table" as the counterpart of the "partitioning tree" as shown in Fig. 1(a). If one chooses f_1 as the first feature all the 1's (patterns (3) and (4)) will be grouped on the left hand side and all the 0's (patterns (1),(2),(5),(6)) will be grouped on the right hand side. Furthermore, one may designate the right hand side as the "right free permutation zone" because any permutations for the patterns (1),(2),(5),(6) is not going to make any difference as far as the classification is concerned. It is easy to recognize a "free permutation zone" since there exist no vertical lines within the "free permutation zone." In this context, the end of the partitioning and classifying condition is reached, when all the free permutation zones consist of no more than one element. When the free permutation zones reach the terminal condition (containing only one element) before the complete classification terminates, their single elements are replaced by a "hyphen" which means "don't care."

Suppose we take the same input matrix just mentioned above, there are three possible ways of partitioning the matrix as shown in Fig. 1(a). However, once we decide to choose f_i as the first feature, then the classification follows the middle path, marked with a heavy line from the root node. Again, there are three possible directions of departure depending upon which feature has been chosen from the remaining set of features. If this process continues, one will be able to trace to the least element if the solution exists. It should be noted that it is possible to have multiple paths, but these eventually emerge into a single path at the end. On the other hand, if the solid line does not reach the terminal node, then a completely separated patterns unique solution simply does not exist. It is also clear that the final node (in the middle level) becomes the starting node for a new feature assignment in the sequence. On the left hand side of Fig. 1(a), one may indicate the features being used along the line as being the index of the dual gradic map. The symbol "*" is being used to represent the situation of completion of successful pattern classification.

In Fig. 1(a) and (b), one can adopt the sequence $\{f_i,f_2,f_3,f_4\}$ in his decision tree. Suppose one permutates the features and decides to try the sequence $\{f_4,f_3,f_1,f_2\}$ as the decision sequence, one discovers that nothing has been gained or lost so far as the classification is concerned as shown in Fig. 1(c),(d) even though the decision path is entirely different, as expected. In the next section one shall see that any permutation would not alter the results so far as the solution is concerned.

It is important to observe that the solution of successful classification may not exist because it is highly dependent on the data structure, the MxN input matrix, even when the positive integer M is much greater than the positive integer N. We demonstrate this fact by constructing a simple example as shown in Fig. 1(e) and (f).

III. An Alternative Mathematical Model - Partition of a Vector

Even though it is natural to model the problem as the partition of a number, there are other advantages in modeling the problem as the partition of a vector using the second level as the starting node instead of the first level as the root node of the tree. Again, we shall employ a simple example as shown in Fig. 2(a) as the vehicle for the development. Assuming N is equal to 4 and it is partitioned into [(3,1)] after the first feature f_1 has been used, one has a two dimensional vector and it is to be partitioned further. Let $\underline{V}$ be a vector, one can write a partition $\overline{V}$ on $\underline{V}$, $\overline{V}=[\underline{V}_1,\underline{V}_2,\ldots j\underline{V}_s]$ where $\sum_{i=1}^{s} \underline{V}_i = \underline{V}$, and call $\underline{V}_i$ in $\overline{V}$ a "part" of $\overline{V}$. If necessary, a partition $\overline{V}$ on $\underline{V}$ is denoted by $[\underline{u}_1^{a_1}\underline{u}_2^{a_2}\ldots\underline{u}_t^{a_t}]$ where $\sum_{i=1}^{t} a_i\underline{u}_i=\underline{V}$ and $\underline{u}_1,\underline{u}_2,\ldots,\underline{u}_t$ are distinct parts of $\overline{V}$. This is called the "standard form" of a partition $\overline{V}$ by Narushima.[8] Let $P(\underline{V})$ denote the set of all partitions on $\underline{V}$. For $\underline{u}$ and $\underline{V}$ in $P(\underline{V})$ one defines $\overline{u} < \overline{V}$ iff $\overline{V}$ can be obtained from $\overline{u}$ after possible rearrangement by adding suitable parts. That is, $\overline{u} \le \overline{V}$ iff $\overline{V} = [\underline{V}_i \mid 1\le i\le n)$ and $\overline{u} = [\underline{u}_{ij} \mid 1\le i\le n, 1\le j\le r(i)]$ where $\underline{V}_i = \sum_{j=1}^{r(i)} \underline{u}_{ij}$ for $1\le i\le n$. Under this ordering $P(\underline{V})$ is a poset with the greatest element $[\underline{V}]$ and the least element $[\underline{e}_1^{V_1}, e_2^{V_2},\ldots\underline{e}_n^{V_n}]$ where $\underline{V} = (V_1,V_2,\ldots,V_n)$ and for $1\le i\le n$, $\underline{e}_i$ is the ith unit vector.

Applying the above analysis to the simple example of Fig. 2(b), one may construct the partitioning tree as shown in Fig. 2(a). The detail of the partitioning tree may be explained through Fig. 2(c). Fig. 2(c) represents the development of a specific path of the tree. There are two points that deserve our attention, one is that the original binary value does not alter under the situation that a new feature does not result in any partition for a "free permutation zone;" the second point is that the binary value for the single element contained within a free permutation zone remains unchanged (replaced by a hyphen -) to the very end of the partitioning process.

It is interesting to observe that the termination condition may not be the least element of the poset as contrary to the other method. As shown in Fig. 2(a) the heavy lined paths represent the solution to the problem where the terminal state $[(1,0)^2(1,1)]$ certainly is not the least element of the poset. Since the vector (1,1) can be decomposed only in one way, (1,0) (0,1). It is correct to say the terminal node * can be located in the lowest two levels of the dual gradic map.

IV. Discussions and Concluding Remarks

(1.) Suppose there is a cost associated with each feature for the classifying machine, as shown in Fig. 2(a). Then for a particular combination of features, say $\{f_1,f_2,f_4\}$, any permutation of these features in the sequence does not change the overall cost.

(2) Suppose the set $\{f_1,f_2,f_4\}$ constitutes the solution to the classification problem, then the adding of a new feature, say f_3, does not contribute to the solution of the problem, rather adding more cost to the classification and recognition.

(3) Once the complete dual gradic map and partitioning tree is fully developed, one may look at the nodes marked with "*" and the solutions may be obtained by tracing the paths in reversed direction.

(4) This problem may be generalized in two directions; features may be chosen for each "free permutation zone," and the second is that the features may not be necessarily "binary-partitioned." In either case, the problem will be more complicated.

(5) So far as the optimal solution is concerned, the dynamic programming techniques may be used to advantage. In the simple example of Fig. 2, it is apparent that the set of $\{f_1,f_2,f_4\}$ and $\{f_1,f_4,f_2\}$ are the sought after solutions shown by heavy paths of the partitioning tree.

(6) The two examples presented in this paper actually are quite simple. For a more complicated and realistic problem, one can forsee the difficulties involved so far as the development of a partitioning tree and partitioning table are concerned. Hence, the development of theoretical analysis presented in this paper set the stage for an efficiency algorithm to cope with the problem. In the algorithm, what one would need, is on the line of heuristic in nature and the results will be presented in a forthcoming paper which deals exclusively with the algorithm and its evaluation. In most realistic problems, the unique signature analysis and encoding procedure for a large number of patterns or classes calls for a more efficient searching of the solution.

REFERENCES

1. Wang, Paul P., and Hodgkiss, W.S.,Jr., "Feature Extraction on a Finite Set of Binary Patterns," Proceedings, International Symposium on Computers and Chinese I/O Systems, Academia Sinica, Aug. 1973, pp. 183-194.

2. Fu, K.S., Min, P.J., and Li, T.J., "Feature Selection in Pattern Recognition," IEEE Transactions on Systems Science and Cybernetics, SSC-6, No. 1, Jan. 1970, pp. 33-39.

3. Levine, M.D., "Feature Extraction: A Survey." Proceedings of the IEEE, Vol. 57, No. 8, Aug. 1969, pp. 1391-1407.

4. Wang, Paul P., and Shiau, R.C., "Machine Recognition of Chinese Characters via Transformation Algorithms," Journal of Pattern Recognition, Pergamon Press, 1973, Vol. V, pp. 303-321.

5. Henrichon, Ernest G.,Jr., and Fu, King-Sun, "A Nonparametric Partitioning Procedure for Pattern Classification," IEEE Transactions on Computers, Vol. C-18, No. 7, July 1969, pp. 614-624.

6. Meisel, William S, and Michalopoulos, Demetrios A., "A Partitioning Algorithm with Application in Pattern Classification and the Optimization of Decision Trees," IEEE Transactions on Computers, Vol. C-22, No. 1, January 1973, pp. 93-103.

7. Stanfel, L.E., "Tree Structures for Optimal Searching," Journal of the ACM, Vol. 17, No. 3, July 1970, pp. 508-517.

8. Narushima, H., "Order Maps and Cardinal Congruence Classes on Partitions," Proceedings of the Faculty of Science, Tokai University, Tokyo, Japan, March 1974, pp. 1-13.

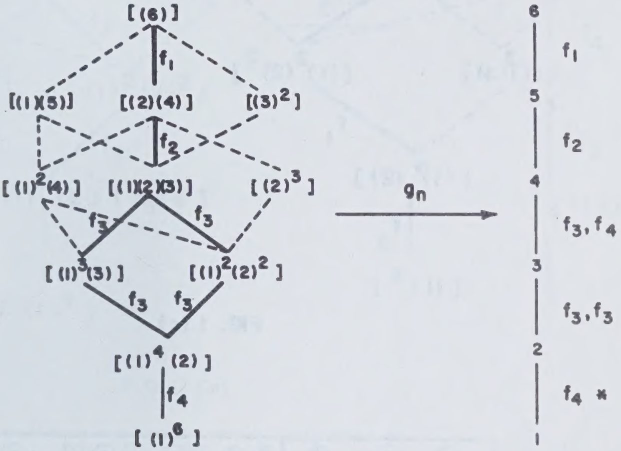
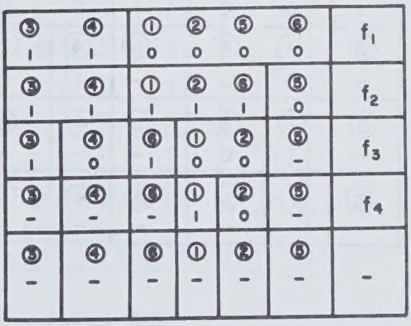
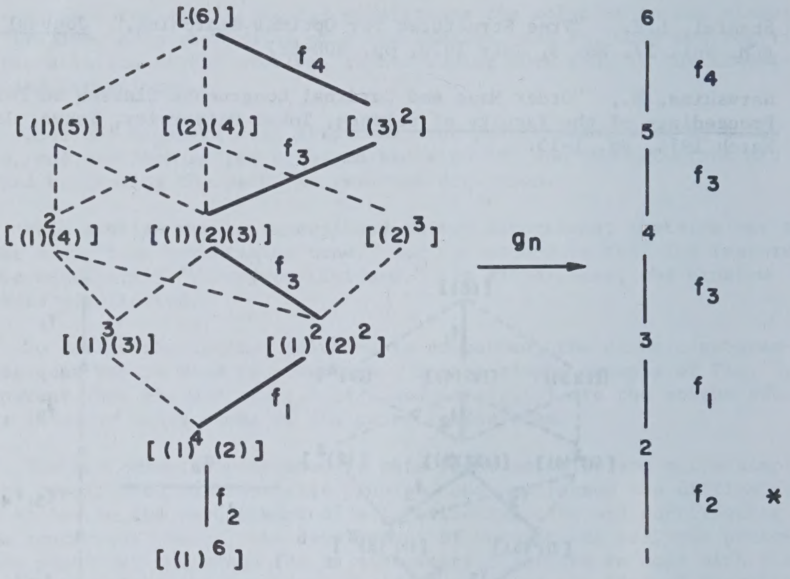

[(6)]
f1
[(1)(5)]
[(2)(4)]
[(3)^2]
f2
[(1)^2(4)]
[(1)(2)(3)]
[(2)^3]
f3
f3
[(1)^3(3)]
[(1)^2(2)^2]
f3
f3
[(1)^4(2)]
f4
[(1)^6]
g_n
6
f1
5
f2
4
f3, f4
3
f3, f3
2
f4 *
1

FIG. 1 (a)



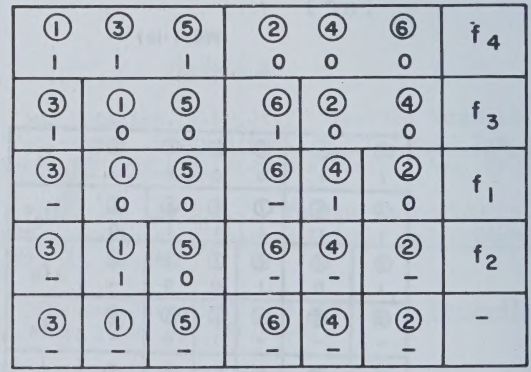
③ 1	④ 1	① 0	② 0	⑤ 0	⑥ 0	f1
③ 1	④ 1	① 1	② 1	⑥ 1	⑤ 0	f2
③ 1	④ 0	⑥ 1	① 0	② 0	⑤ –	f3
③ –	④ –	⑥ –	① 1	② 0	⑤ –	f4
③ –	④ –	⑥ –	① –	② –	⑤ –	–

FIG. 1 (b)



[(6)]
f_4
[(1)(5)]
[(2)(4)]
$[(3)^2]$
f_3
$[(1)^2(4)]$
[(1)(2)(3)]
$[(2)^3]$
f_3
$[(1)^3(3)]$
$[(1)^2(2)^2]$
f_1
$[(1)^4(2)]$
f_2
$[(1)^6]$
gn
6
f_4
5
f_3
4
f_3
3
f_1
2
f_2
*
1

FIG. I (c)



① 1	③ 1	⑤ 1	② 0	④ 0	⑥ 0	f_4
③ 1	① 0	⑤ 0	⑥ 1	② 0	④ 0	f_3
③ –	① 0	⑤ 0	⑥ –	④ 1	② 0	f_1
③ –	① 1	⑤ 0	⑥ –	④ –	② –	f_2
③ –	① –	⑤ –	⑥ –	④ –	② –	–

FIG. I. (d)

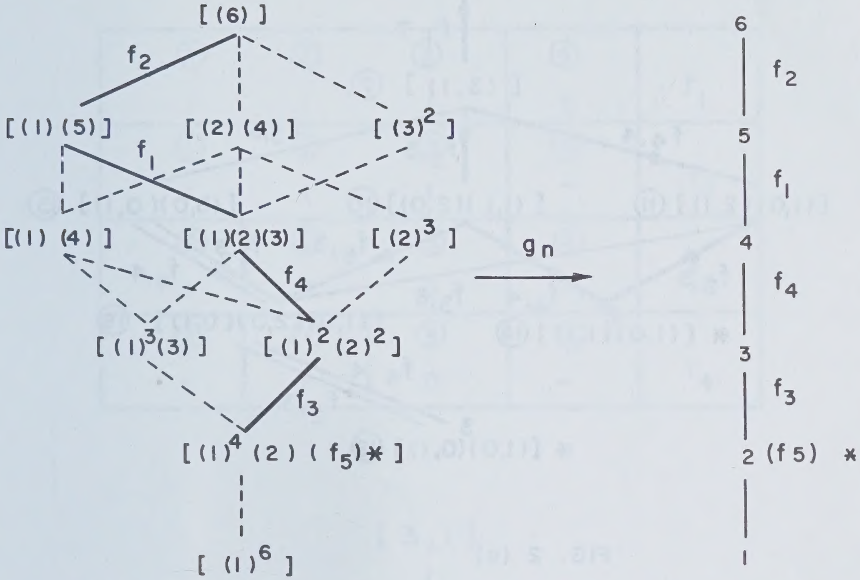

[(6)]
f_2
[(1)(5)]
[(2)(4)]
[$(3)^2$]
f_1
[(1)(4)]
[(1)(2)(3)]
[$(2)^3$]
f_4
[$(1)^3(3)$]
[$(1)^2(2)^2$]
f_3
[$(1)^4$ (2) (f_5)*]
[$(1)^6$]
g_n
6
f_2
5
f_1
4
f_4
3
f_3
2 (f5) *
1

FIG. 1 (e)

① 1	② 0	③ 0	④ 0	⑤ 0	⑥ 0	f_2
① –	② 1	③ 1	④ 0	⑤ 0	⑥ 0	f_1
① –	② 1	③ 1	④ 1	⑤ 0	⑥ 0	f_4
① –	② 1	③ 0	④ –	⑤ 0	⑥ 0	f_3
① –	② –	③ –	④ –	⑤ 0	⑥ 0	f_5

FIG. 1 (f)

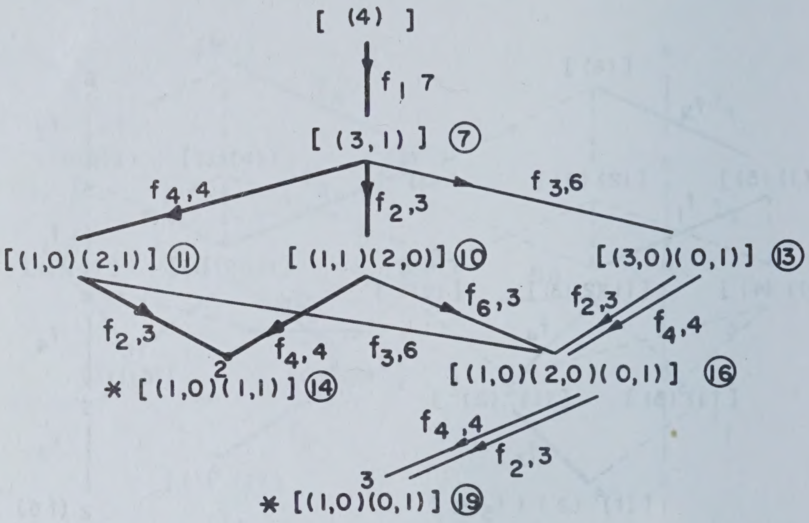
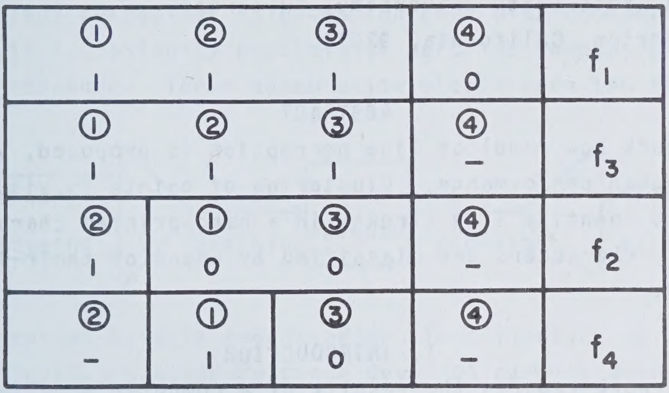

[(4)]
f1 7
[(3,1)] ⑦
f4,4
f2,3
f3,6
[(1,0)(2,1)] ⑪
[(1,1)(2,0)] ⑩
[(3,0)(0,1)] ⑬
f2,3
f6,3
f2,3
f4,4
2
f4,4
f3,6
[(1,0)(2,0)(0,1)] ⑯
* [(1,0)(1,1)] ⑭
f4,4
f2,3
3
* [(1,0)(0,1)] ⑲

FIG. 2 (a)

1	2	3	4	
1	1	1	0	f_1
0	1	0	1	f_2
0	0	0	1	f_3
1	0	0	0	f_4

FIG. 2 (b)


① ② ③ ④
1 1 1 0
f1
① ② ③ ④
1 1 1 –
f3
② ① ③ ④
1 0 0 –
f2
② ① ③ ④
– 1 0 –
f4

$$\begin{gathered}[3,1] \\ \downarrow \\ [(3,0)(0,1)] \\ \downarrow \\ [(1,0)(2,0)(0,0)(0,1)] \\ \downarrow \\ [(1,0)(0,0)(1,0)(1,0)(0,0)(0,0)(0,0)(0,1)] \\ =[(1,0)^3(0,1)]\end{gathered}$$

FIG. 2 (C) PARTITIONING TABLE

A CHARACTER PERCEPTION-RECOGNITION SYSTEM

B. E. GALA AND D. A. MICHALOPOULOS
Department of Quantitative Methods
California State University, Fullerton
Fullerton, California, 92634

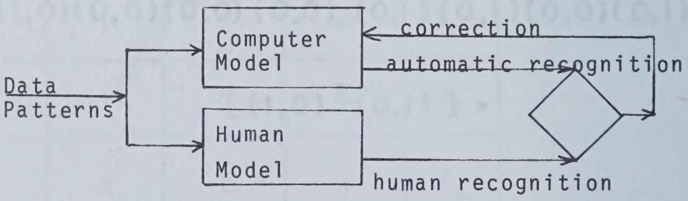
ABSTRACT

A black box model of line perception is proposed, which matches human performance. Clustering of points in visual space is used to identify line strokes in a hand-printed character data set. Characters are classified by means of their line structure.

I. INTRODUCTION

This work reports the results of a computer modeling approach in the investigation of human perceptual processes. We hypothesize a model of perception and a computer program is written which tests the hypothesis against visual data and human performance. If a program feature improves the recognition rate of the machine, it is maintained; if not, it is deleted from the model. Sometimes the entire computer program is rejected and a new model is coded for testing.

We have been describing an adaptive system identification process with man in the feedback loop, as shown in the figure below.

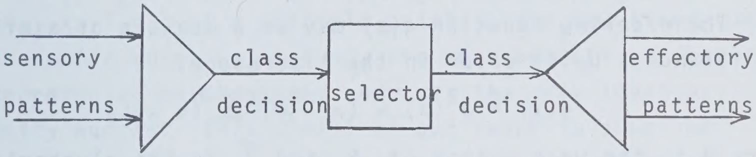


Eventually a machine model of perception may emerge that matches human recognition performance. The underlying mechanisms of the machine program may correspond with those of human perception. If this is the case, the model will fit data obtained from neural probes and from biological and psychobiological methods, including histological findings.

II. PERCEPTION AND UNSUPERVISED ESTIMATION

What is Perception?

If we show a card with the letter A on it to a subject and he responds by reading "A", we say that he has perceived the letter. Thus perception is often confused with recognition; however, it is obviously possible to perceive something without making a response. The diagram below should make the situation clear.



Perception is data reduction or classification up to the selector level. A motor response does not have to emerge. Learning can occur in perception, but it is generalized learning not the conditioned-response kind. The perceptual process reduces sensory data into manageable groupings. Significant features emerge from these groupings and on the basis of past conditioning, in feature space, an organism makes an appropriate response.

A valid model of perception must work for all signals, not only for conditioned stimuli. This is what is known in the pattern recognition field as unsupervised classification. In some cases, data may be organized on the basis of a criterion or objective function and an algorithm or a process. Two approaches that have been studied in unsupervised learning or estimation are identification of finite mixtures -- a theory with little practice; and clustering -- a practice with little theory.

A Clustering Transformation

Good reviews of clustering techniques are given by Ball[1] and more recently by Anderberg [2] .

Let X be a set of vectors in R^m space, g(x) an ordering function on R^m and t a real—valued threshold.

Definition: A cluster $B \subset X$ is defined by the relation $B = \{x : g(x)\} < t$.

Clustering involves the iteration of a set of sample points

$\{x_i\}_{i=1}^{n}$ at least once, through decision rules $g_k(x) < t_k$, $k = 1,2,\ldots K$. K is the number of clusters or classes. This process results in the assignment of X to different groupings $\Omega_1, \Omega_2, \ldots, \Omega_j, \Omega^*$. Each Ω is one of K^n possible combinations; a point in n-dimensional space of K quanta along each dimension. The final grouping Ω^* may minimize some objective function on the points X: $J = (\Omega, X)$.

The ordering function g(x) may be a measure of similarity or distance. Quite often in the literature,

$$g(x) = d^2(x) = (x - \mu)^t \Sigma^{-1} (x - \mu) \qquad (1)$$

where μ is the mean vector of B and Σ is the cluster's covariance matrix, iteratively computed. Let $y = x - \mu$. If Φ is the eigenvector matrix and Λ the eigenvalue matrix of Σ, then $\Phi\Sigma\Phi = \Lambda$ or $(\Phi \Sigma \Phi)^{-1} = \Lambda^{-1}$ or $\Sigma^{-1} = \Phi^t \Lambda^{-1} \Phi$, using the fact that $\Phi^{-1} = \Phi^t$ due to the orthogonality of Φ. Equation (1) becomes $d^2(y) = (y^t \Phi^t \Lambda^{-\frac{1}{2}})(\Lambda^{-\frac{1}{2}} \Phi y)$ or $d^2(y) = y^t(\Phi^t \Lambda^{-\frac{1}{2}})(\Lambda^{-\frac{1}{2}}\Phi)y$. (2)

The matrix $T = (\Lambda^{-\frac{1}{2}}\Phi)$ in Equation (2) is the whitening transformation. T uncorrelates the data and changes the scale of the principal components by normalizing with the standard deviations $\sqrt{\lambda_i}$.

Let $z = Ty$, then Equation (1) becomes

$$d^2(z) = z^t z = \|z\|^2 \qquad (3)$$

If the factor $e = (\prod_{i=1}^{m} \lambda_i)^{\frac{1}{m}}$ is included in the transformation, it can be shown that T minimizes $\overline{d^2}$, the mean-square intra-cluster distance, subject to a constant volume constraint $\det T^{-1} - 1 = 0$. (Sebestyen [3])

Equation (1) appears also in the exponent of the m-variate Gaussian probability density. It is a hyperparaboloid of revolution and respresents the quadratic loss of information in the process of clustering. the cross-sections of $d^2(x)$, $d^2(x) = t$, are hyper-ellipsoids whose size increases with t. The hyper-ellipsoids have semi-axes whose directions are defined by the eigenvectors of Σ. This restricts the decision regions to convex volumes. The length of each semi-axis is $\sqrt{t\lambda_i}$ for each dimension i. It is apparent that fitting a Gaussian density to the cluster B is equivalent to clustering with $d^2(x)$.

III. A CLUSTERING ALGORITHM FOR PERCEPTION

Image Clustering

We can define any black and white image as a matrix of two-tuples which represent the coordinates of retinal events:

$$X = (x_{11},x_{21}),(x_{12},x_{22}),\ldots,(x_{1n},x_{2n})$$

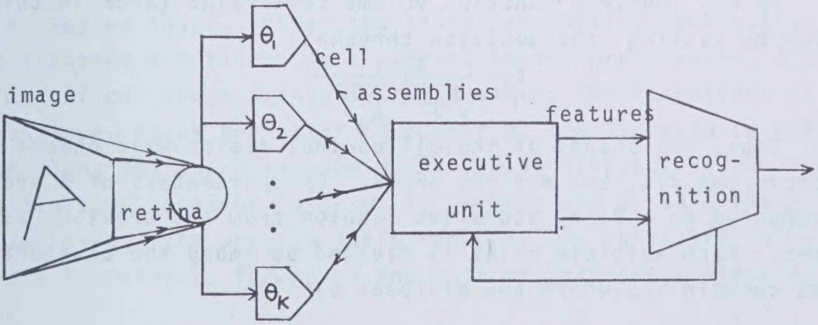
By collecting events with the same coordinates, we can compress X into the set of ordered pairs

$$X = [(x_{11},x_{21}),\ c_1],[(x_{12},x_{22}),c_2],\ldots,[(x_{1n},x_{2n}),c_n]$$

where $[(x_{1i},x_{2i}),c_i]$ is a multiple point in the two-dimensional image space, for which x_{1i} and x_{2i} are the vertical and horizontal coordinates respectively and c_i is the gray level or multiplicity number. This X will be our input to a set of decision rules d(x) and it will be convenient to refer to a cluster B as a collection of such multiple points.

Zahn [4] has experimented with the use of the minimal spanning tree for cluster detection, taking into account relations (distances) between some of the points in B, those distances that form the MST. Our algorithm, using $d^2(x)$, utilizes all relations in B, the entire graph, and from the statistics of the cluster, extracts shape information.

The algorithm is intended for parallel processing and our perceptual unit is one of a number of identical parallel units that estimate the cluster parameters θ_k for k=1,2,...,K. We may think of these units as the cell assemblies postulated by D. O. Hebb [5]. A cell assembly is a diffuse collection of communicating neurons that incorporates the engram of a particular concept. In our problem, the cell assembly stores the values of θ_k, defines the field or area of the cluster in the image, by means of $d^2(x)$, and communicates with a central or executive unit as is shown schematically below.



As the visual fields of the units change and grow with time (iterations of X through the decision rules), many cell assemblies become redundant and are decoupled; this way, processing is speeded up. At the end, from one to four units only remain in the case of printed character recognition.

A Step-by-Step Description of the Clustering Algorithm.

Details of each step of the algorithm can be found in [6].

Step 1. Generate initial cluster centers. This step is quickly performed by the executive unit. Cluster centers are distributed uniformly in the regions of the image space where points exist. The spacing of the centers depends on the number of available cell assemblies or computing power. For high resolution pictures a large number of centers must be generated. For our purpose of character recognition, ten centers proved sufficient.

Step 2. Estimate parameters of local clusters using linear model. Consider the relation

$$d^2(x) = [\Phi(x-\mu)]^t \Lambda^{-1} [\Phi(x-\mu)] < t \quad (4)$$

For m=2, Equation (4) represents an ellipsoidal data window or surface. The area of the ellipse (the visual field of the cell assembly) is $\pi t\sqrt{\lambda_1\lambda_2}$. If $\lambda_1>\lambda_2$, $\sqrt{t\lambda_1}$ and $\sqrt{t\lambda_2}$ are the major and minor semiaxes of the ellipse respectively. Before we can iterate X through (4), we must set up a lower bound for the eigenvalues so as to avoid a collapse of the cluster B in the event the points of B for a line. The major eigenvector of Φ gives the coordinates of the straight line which best fits B, in the sense of miniming the mean-square distance of B from the line and thus we obtain a first approximation of the line strokes making up the printed character.

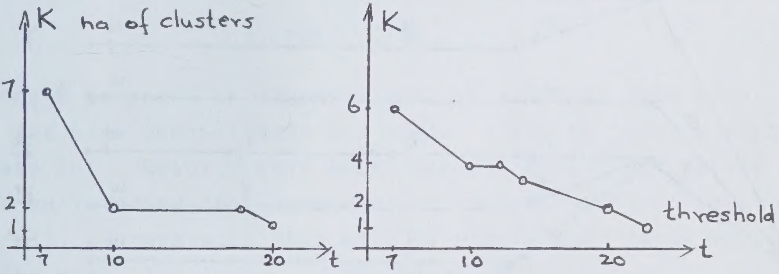
We can impose a constant volume constraint (area in this case) by setting the decision threshold

$$t_j = \frac{t_{j-1}}{\sqrt{\lambda_1^{j-1}\,\lambda_2^{j-1}}}$$

Thus, the shapes of the ellipsoidal fields will change to improve the fit, but not the areas. The parameters of B are recomputed in a fast, iterative fashion from the previous estimates. Each multiple point is divided up among the clusters that contain it, where the ellipses overlap.

Step 3. Elimination of Redundant Cell Assemblies. This step occurs after every iteration of X and handles the minimization of K (the number of perceptual aggregates or clusters) in a vector criterion $J = (A,K)$, where A is the mean-square intra-cluster distances sum. A count of the points in each cluster that are shared with another cluster is kept during the iteration of X and if the count exceeds an engineering parameter the cell assembly for the smaller cluster is decoupled.

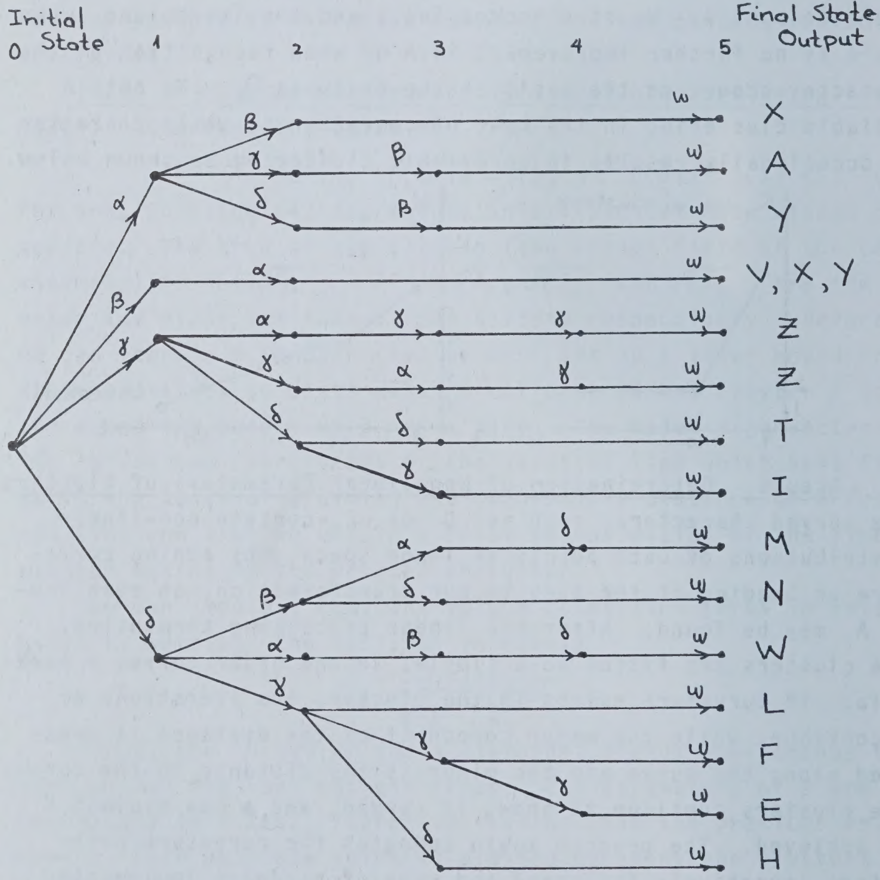
Step 4. Determination of Decision Thresholds. For the minimization of K to work, we must enlarge the data windows or ellipsoidal areas so that elimination can occur by overlap as in step 3. This is done by incrementally increasing the threshold t and then going through one or more constant volume iterations of X. We stop increasing t and the iterations, when there is no further improvement in A or when recognition of the character occurs on the basis of the features θ_k. We obtain reliable clustering in the case of character L, while character M occasionally results in unreliable clustering as shown below.



Step 5. Determination of Non-linear Parameters of Clusters The curved characters, such as D or C, contain non-linear distributions of data points in image space. By adding curvature or bending of the axes to our transformation, an even lower A may be found. After the linear processing terminates, the clusters are fitted to a simple, second order curve, a parabola. If curvature exists in the cluster, the iterations of X continue, while the major component of the distance is measured along the curve and the minor is the distance to the curve. The clusters continue to grow, if curved, and a new minimum K is achieved. The program again computes the curvature parameters iteratively for speed and ease of hardware implementation.

IV. HAND-PRINTED CHARACTER RECOGNITION

Once feature extraction of the line strokes has resulted from the clustering process, the strokes are classified on the basis of their slope, length and curvature. The classified strokes can now be called picture primitives after Shaw [7] and other workers in the field of artificial intelligence and linguistics. The primitives are submitted to a finite state recognition automaton, whose output gives the recognition of the printed character. For example, if we classify straight line strokes by means of the slope feature alone and we let the primitives be α for the / stroke, β for \, γ for — and δ for |, adding ω for the "end-of-string" primitive, we obtain the finite state recognition automaton shown below.



We have conducted numerous computer experiments of the clustering algorithm using a data set supplied to us by Recognition Equipment Corporation of Dallas, Texas. This data set is derived from FORTRAN coding sheets, generated by a variety of persons and characters have considerable noise and distortion. Each isolated hand-printed character has been exposed to an array of photocells which digitize the image into 192 (12 x 16) three-bit numbers representing the gray level. White is coded as 0 and black as 7.

Our clustering algorithm, while simulating 10 parallel units on a digital computer, was slow, taking up at least one minute on the IBM 360/44 computer, in order to decompose each printed character into line strokes. Performance with regard to straight line strokes was very nearly perfect. Curved characters were not always decomposed into the same primitives as by a human reader. We also programmed the simple recognition automaton of the preceding figure and the error rate was very nearly zero for straight-line characters.

V. CONCLUSION

We have proposed an adaptive control model of line perception and have demonstrated its basic validity using a realistic data set. Results have been encouraging and our belief has been re-inforced that eventually a control model of perception will be generated that will be widely applicable in processing most line images of practical interest.

However, a structural analysis of line pictures is not practical in cases of high noise, although the clustering process has some noise-rejection capability. In fact, clustering need not stand alone in feature extraction or recognition. Correlation and context may be employed simultaneously for more perfect recognition so error rate of human readers may be reached.

Present-day optical recognition equipment (OCR) employs local masks in fixed regions of the visual space for the extraction of features and correlation for recognition. Clustering, which is independent of the location and orientation of the line stroke, will add a new and important dimension to the capabilities of these machines. If clustering turns out to be a slow process when actually implented in hardware, perhaps it will be used only when high correlation cannot be obtained for unknown characters.

REFERENCES

1. G. H. Ball,"Data Analysis in the Social Sciences: What about the Details?" in 1965 Fall Joint Comput. Conf. AFIPS Conf. Proc.,vol.27,part I,pp.553-559, Washington, D. C., Spartan Press, 1965.

2. M. R. Anderberg, "Cluster Analysis for Applications," Kirtland AFB, New Mexico, January 1972 (AD 738301).

3. G. S. Sebestyen, "Decision-making Processes in Pattern Recognition," pp.25-30, Macmillan Co.,1962.

4. C. T. Zahn, "Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters," IEEE Trans. Comp.,vol.C-20, No. 1, pp.68-86, January 1971.

5. D. O. Hebb, "The Organization of Behavior," John Wiley and Sons, Inc., N. Y., 1959.

6. V. E. Galanoudes (B. E. Gala), "An Adaptive Control Model of Line Perception Applied to the Design of a Character Recognition System," PhD Dissertation, University of Southern California, June, 1973.

7. A. C. Shaw, "A Formal Picture Description Scheme as a Basis for Picture Processing Systems," Information and Control, vol. 14, pp.9-52, January 1969.

A SIMPLE CORRELATION METHOD FOR HANDWRITTEN CHARACTER RECOGNITION

MICHAEL E. IRRGANG and STEPHEN S. YAU
Department of Computer Sciences
Northwestern University
Evanston, Illinois 60201

ABSTRACT

A new approach to feature extraction for hand-printed character recognition was examined. This approach treated characters as binary images rather than line drawings. The feature extraction consisted mainly of stylizing the letters such that all lines were of single-point thickness and either vertical, horizontal, or diagonal. The actual recognition process can then be a simple correlation.

This approach was tested using a standard data base. It was found that a large degree of success will be achieved if moderately well written data, especially with characters which have most of their major features around their periphery, is used.

I. INTRODUCTION

In recent years much work has been done in the field of hand-written character recognition. Though many of the algorithms used have had high recognition rates, most of them have required much computation and also a large scale general purpose computer in order to attain any acceptable speed of recognition. We have attempted to develop a feature extraction algorithm which is relatively fast; which does not require a large scale computer, but instead has been implemented on a DEC PDP-9, and which enables recognition by a simple correlation.

II. PRELIMINARIES

In character recognition, there are two basic approaches to feature extraction. A character can be treated as a line drawing, which is probably what the human brain does, or it can be treated as a binary picture. This latter is generally the approach used for machine print recognition.

We must first consider the problems in recognition of machine printed characters. Most material to be processed today by OCR [Optical Character Recognition] is a combination of typing and computer output. The latter is often fuzzy, and usually misaligned either vertically or horizontally, depending on whether a chain or a drum printer was used. Missing or uneven dots become the worst problem when using a matrix printer. Typing tends generally to be of better quality, but even more misaligned and with more noise in the form of smudges.

The earliest work in OCR concentrated on the recognition of single fonts. For this type of recogntion the simplest method is optical template matching, in which one has a template of the ideal character and attempts to superimpose it over the test pattern. It is, of course, necessary to have a positive as well as a negative template, since, for example, an E would entirely cover a negative template for an F.

Later the optical template became more generalized with the development of a weighted gray template[1]. This was not really a template, but an optical correlation matrix. The idea of an optical correlation matrix was applied to digital techniques which were used in late multifont recognition machines.

Most work in recent years has been some form of processing binary matrices. A 20 x 20 matrix, plus or minus 4 or so, has usually been found necessary and sufficient for recognition purposes. Some preprocessing is usually done to filter out noise and fill in gaps. Various recognition methods then follow. One popular method is peephole template matching [7], in which just a few points are used which have a maximum weight for recognition. This is also called N-tuple matching[3].

Hand print recognition is an outgrowth of machine print recognition. In dealing with handwritten characters there is usually less noise in the form of stray marks than one finds when dealing with machine print. The problems one does encounter are quite numerous, however (examples: broken lines and gaps, wide variation in size, skewed characters, difficulty in finding the next character). Much work has been done in attempting to read the alphanumerics, but recognition is generally no better than 75 to 80%. It is important to note that, when dealing with handprinted characters, the deviation from the standard becomes much greater, and the definition of a character much fuzzier, than with machine printed characters. It is then usually necessary to constrain the writers, requiring them to write within a box, and conform to a set of rules for neatness.

In recognizing hand printing, templates or correlation matrices alone are almost impossible to use in an unconstrained situation. Even when letters are constrained, some normalization is necessary. One interesting type has been normalization by moments. This is excellent pre-processing for any subsequent recognition method [8], but the computation time is severe. One interesting recognition method is that of characteristic loci [6] The principal interest of this method is that it requires no normalization, but percent recognition is not high.

The methods mentioned so far treat characters as pictures. Methods which treat characters as line drawings deal with curve following, and recognition by means of analysis of the number of corners, intersections, their locations, etc. Such methods would be very good, except that they often break down because of their high sensitivity to breaks in lines or noise. They are also generally quite expensive in processing time.

One of the earliest studies using the above methods, [2], rather than examining a binary matrix of a letter, examined the graph of position versus time in a curve following scan of the letter. Normalization, therefore, was of the duration and amplitude of this graph. The problems were those of any of the curve following methods, however.

A very successful use of these methods was in [4], where the letters were input to the computer as they were being written, or drawn. The computer was therefore able to follow the curves as they were drawn, which resulted in a very high recognition rate. This procedure is typically used with tablet input.

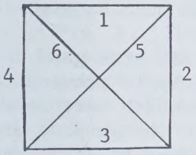
One of the most recent works [10] was an examination of the topological features occurring in the different octants of a circle. Normalization of size is unnecessary. The recognition rate was quite good, 94%, but all input was by light pen, which is rather untypical. Recognition was a two stage process: first the octant method, and then a good deal of subsequent processing to break down the subgroups. The method required a great amount of computation, and has the further disadvantage of being very difficult to transfer to a different character set.

Most of the work in the field of hand print recognition has been recognition of line drawings. Usually only in machine print recognition were algorithms used which instead were processes of recognizing simple binary images. An algorithm of this lattertype can be considerably simpler, and this is what we have attempted to use for hand printed characters.

III. THE APPROACH

3.1-The Recognition Algorithm

1. Input an area containing a character into an $n \times m$ binary matrix, where $n,m \geq 16$. Eliminate any point which has fewer than two neighbors, assuming it to be positive noise.
2. Now examine the matrix row by row and column by column from all four edges inward to determine which are the outermost rows and columns containing a point. Using these limits, calculate boundaries for a size and position normalized letter for a rescan of the original matrix. Move the character into a 20 x 20 binary matrix, by scanning the original matrix between these boundaries. Eliminate negative noise by adding points for holes with six or more neighbors. This will close some small gaps.
3. Perform the thinning algorithm (Appendix) until completion.
4. Count the number of points in the letter matrix with only one neighbor to get the number of endpoints.
5. Squeeze all four edges of the letter inward two places, reducing the letter from 20 x 20 to 16 x 16. This straightens the lines along the periphery of the character, making it appear more rectangular.
6. Divide the 16 x 16 letter into 16 4 x 4 submatrices. In each submatrix, by summing the squares of the distances for all points in the submatrix, find which of the following lines is most closely approximated, and assign the submatirx a value which is the number of that line:



   If the submatrix has fewer than three points, it is considered empty and is assigned the value 7.
7. The values assigned to each submatrix form a 4 x 4 feature matrix for the letter. Also associated with the letter is its number of endpoints, as determined in Step 6.
8. Correlate the feature matrix to each of a set of average character matrices. Ignoring all classes which do not allow the number of endpoints possessed by this character, find the maximum of these correlations. This is the class to which the character belongs.

Figure 1 is an example of the complete feature extraction process. The step numbers of Algorithm 3.1 are in brackets.

3.2-The Selection of Matrix Size

It should be noted that the development of any new character recognition algorithm is purely a heuristic process. That is, there exist no sound theoretical grounds for assuming that any given idea is going to work better than any other. In choosing the matrix size there were two things to consider. It was desirable to have a high recognition rate, but it was also important not to increase the computation time. Going from 20 x 20 to 24 x 24 would involve a 45% increase in computation time. It has previously been stated that computation time was the most important factor.

The minimum size for the feature matrix was considered to be 4 x 4. Characters of the alphanumeric set generally consist of line segments around the periphery of a rectangle or through the center of a rectangle. A loop in a character will generally occupy one half of that character. In fact, any major feature of a character will occupy half of the character. Therefore, the character can be described by describing the upper and lower halves. Each half can be stylized and represented by a 2 x 4 matrix of the line segments as described in Algorithm 3.1. However, it is now clear that a normalized character should be a square matrix, and that its dimensions must be a multiple of 4. 20 x 20 is felt to be the minimum size for the character matrix, due to the "squeezing" and the minimum size of 4 x 4 for the set of 16 submatrices, as discussed in Section 3.4.

3.3-Input and Normalization

Some form of normalization upon input is a requirement in most character recogntion systems. Moment normalization is a particularly good method of standardizing characters, but unfortunately it requires a considerable amount of computation. It was felt that merely normalization of size and position, a simple iterative process, would produce results almost as good but in a much shorter time.

The size of the input matrix before normalization is unimportant. It should merely be greater than the 16 x 16 size of the thinned character matrix. The boundaries of the character in the original matrix are used to find a delta x and delta y for the normalization rescan. Since the total difference is unlikely to be a multiple of 20, these deltas will be floating point numbers used in floating point increments of the array indices.

The final noise elimination fills some gaps in lines and smoothes smearing of edges. This is necessary to prevent the line thinning algorithm from leaving stray lines which were merely noise-caused breaks in the contours of the characters. The purpose of the earlier noise elimination, before normalization, was to eliminate stray marks, which will frequently occur if, for instance, the input device is a flying spot scanner.

3.4-Generation of the Feature Matrix

Assuming that a character is to be described by a 4 x 4 feature matrix of the line segments as stated in the algorithm, it was felt that each of the line segments should be obtained by fitting the points in approximately a 4 x 4 submatrix to the nearest of the line segments, using a least squares fit. A character thus had a 20 x 20 representation. It was found, however, that due to application of the thinning algorithm, those characters which originally consisted to thick lines were now somewhat de-normalized. A very simple solution to this problem was to "squeeze" in the edges of a 20 x 20 character uniformly two rows (and columns) toward the center, to form a 16 x 16 character.

3.5-Preclassification by Endpoints

The word "preclassification" used here means that certain classes are eliminated as possibilities for a character. In other words, for a given character, the number of choices in its final classification is reduced.

Due to the simplicity of our algorithm, it needs one slight addition before it is able to distinquish between similar classes. This addition is pre-elimination of classes by the number of endpoints. At the end of the thinning algorithm, any endpoint of a line will have exactly one neighboring point. The endpoints are counted, and those classes which could not possibly have that particular number of endpoints are not considered in the subsequent correlation classification. It was felt that an E with three endpoints should be considered equally as likely as one with eight endpoints. On the other hand, a character with zero endpoints could not be an E. Even if it could have correlated to E it is probably a B. After pre-elimination of classes by endpoints, another process involving little computation, recognition is performed using the remaining classes by correlation of the feature matrix to a set of average letters.

IV. THE EXPERIMENTS

To test our algorithm, several experiments were conducted. It was felt that it would be of value to use a standard data base [5]. This data base consisted of 147 alphabets of the 46 FORTRAN characters. Of these characters only A-Z and 0-9 were used. The data base for this experiment thus consisted of 147 alphabets of 36 characters, or 5292 characters. The 36 different characters were treated as 40 pattern classes: G and G ; J , J , and J ; and U and U were considered to be distinct classes.

The characters were represented by 24 x 24 binary matrices, and were originally input with a vidicon scanner. According to [5] the vidicon produces little noise in the form of scattered dots, but does produce considerable "smearing". In all experiments the sample patterns were also used for test patterns. The results of these experiments are tabulated in Table 1, and the average recognition rates as a function of the number of classes are shown in Fig. 2. These recognition rates are for both 20 x 20 correlation and for Algorithm 3.1.

In the experiments on the entire data set, it is shown that our algorithm does not recognize average untrained data very well. Compared with [6], which used the same data and a recongition method of Characteristic loci, the recognition rate of our algorithm is about 11% lower for A-Z and about 17% lower for 0-9. Our feature extraction algorithm is quite sensitive to any loops which may have been filled in and to any open curves which become closed due to the line thinning process and the noise elimination procedure. The SRI data contains many characters with both these problems, not only because of poorly written characters, but also because of the "smearing" effect of the vidicon scanner used as the input device. Typical examples of characters which often exhibit such problems are G's and J's with closed curves and 8's with filled in loops.

In the subsequent experiments we try to show how our algorithm works with data which does not suffer from the above problems. In comparison with the direct correlation recognition, our algorithm has a more than 10% higher average recognition rate. More important, however, is the fact that our algorithm is more consistent in its results. The second part of this experiment, the graphs of Rate=F(# of classes), also showed that our algorithm is quite reliable and useful.

From examination of the classes which were at the extremes in recognition rates, it is possible to draw certain conclusions regarding the abnormally high or low recognition rates observed. Any two characters which are similar, except that one has sharp and one has rounded corners, will be considerably confused. Such was the case with D and O, B and 8, and F and P. The confusion is caused by the thinning of lines followed by squeezing. The corners of a rounded character will always fit to a diagonal line, whereas a square character will do so only part of the time. Therefore, the resultant correlation matrices will cause any character which has the diagonal corner fit to be classified as the rounded character. Characters with no closed loops generally had high recognition rates. The method also breaks down when dealing with interior features of characters, giving the best results for classes with the major features on the periphery. Finally, characters which contain acute angles generally had poor recognition rates. This again is due to the line thinning, which can turn an acute angle into either a straight line or an actual angle.

V. CONCLUSION

In this paper a new approach to character recognition, or, rather, to feature extraction of characters, has been described. The feature extraction process attempts to stylize and simplify the characters, using a simple algorithm which involves little computation. The recognition is then done by correlation to "average letter" correlation matrices. The method was tested using a standard data base [5], and showed a moderately good recognition rate. Though the method was used only for alphanumerics, it could be used for any simple characters consisting of two to four line segments or curves.

The recognition rates for the standard data base were rather low. This is largely because the major emphasis in the development of this algorithm was on simplicity and speed of computation, and as a result, it is not as effective an algorithm as some other, more sophisticated ones which require more computation time. We compared our algorithm with another of comparable simplicity, and ours showed significantly better results.

We were able to classify one character every six seconds on a DEC PDP-9. This time would undoubtedly drop considerably if the algorithm was implemented in hardware. Another aspect of this recognition algorithm is that it consistently classified a character correctly on either its first or second choice. Therefore, this algorithm would also be very effective as a pre-processor for a syntactic recognizer.

An average recognition rate of over 95% could possibly be achieved by using only trained writers. If trained writers were used it would be possible to further constrain the pre-classification by endpoints, limiting the allowed numbers of endpoints for each class.

APPENDIX

Following is the algorithm used for thinning the lines, taken from [9].

The neighbors of a point are labeled as follows:

$$\begin{matrix} p_9 & p_2 & p_3 \\ p_8 & p_1 & p_4 \\ p_7 & p_6 & p_5 \end{matrix}$$

Let $A(p_1)$ be the number of patterns 01 in the ordered set p_2, p_3, p_4, ... , p_9, p_2 of the neighbors of p_1. Let $B(p_1)$ be the number of nonzero neighbors of p_1. Then delete a point from the character if all the following are true:

$$2 \leq B(p_1) \leq 6$$

$$A(p_1) = 1$$

$p_2 \wedge p_4 \wedge p_8 = 0$ or $A(p_2) \neq 1$

$p_2 \wedge p_4 \wedge p_6 = 0$ or $A(p_4) \neq 1$

Reiterate this algorithm until no deletions are recorded for a full iteration. All line are now thinned completely with the contours preserved, and contour noise eliminated.

BIBLIOGRAPHY

In Approximately Chronological Order

1. Highleyman, W. H., "An Analog Method for Character Recognition," IRE Trans. Elec. Computers, EC-10:502-512, (1961).

2. Greanias, E. C., et al., "The Recognition of Handwritten Numerals by Contour Analysis," IBM Journal of Res. and Dev., 7:14-21, (1963).

3. Liu, C. N. and Shelton, G. L., Jr., "An Experimental Investigation of a Mixed-Font Print Recognition System," IEEE Trans. Elec. Computers, EC-15:916-925, (1966).

4. Groner, G. F., "Real Time Recognition of Handprinted Text," Proc. FJCC, 29:591-601, (1966).

5. Munson, John H., "The Multi-Author Hand-Printed FORTRAN Alphabet Data File," Stanford Research Institute (1967).

6. Knoll, A. L., "Experiments with 'Characteristic Loci' for Recognition of Handprinted Characters," IEEE Trans. Computers, C-18:366-372, (1969).

7. Balm, G. J., "An Introduction to Optical Character Reader Considerations," Pattern Recognition, 2:151-166, (1970).

8. Casey, R. G., "Moment Normalization of Handprinted Characters," IBM Journal of Res. and Dev., 14:548-557, (1970).

9. Stefanelli, R. and Rosenfeld, A., "Some Parallel Thinning Algorithms for Digital Pictures," Journal of the ACM, 18(2):255-264, (1971).

10. Tou, J. T. and Gonzalez, R. C., "Recognition of Handwritten Characters by Topological Feature Extraction and Multilevel Categorization," IEEE Trans. Computers, C-21:776-784, (1972).

11. Harmon, Leon D., "Automatic Recognition of Print and Script," Proc. IEEE, 60:1165-1176, (1972).

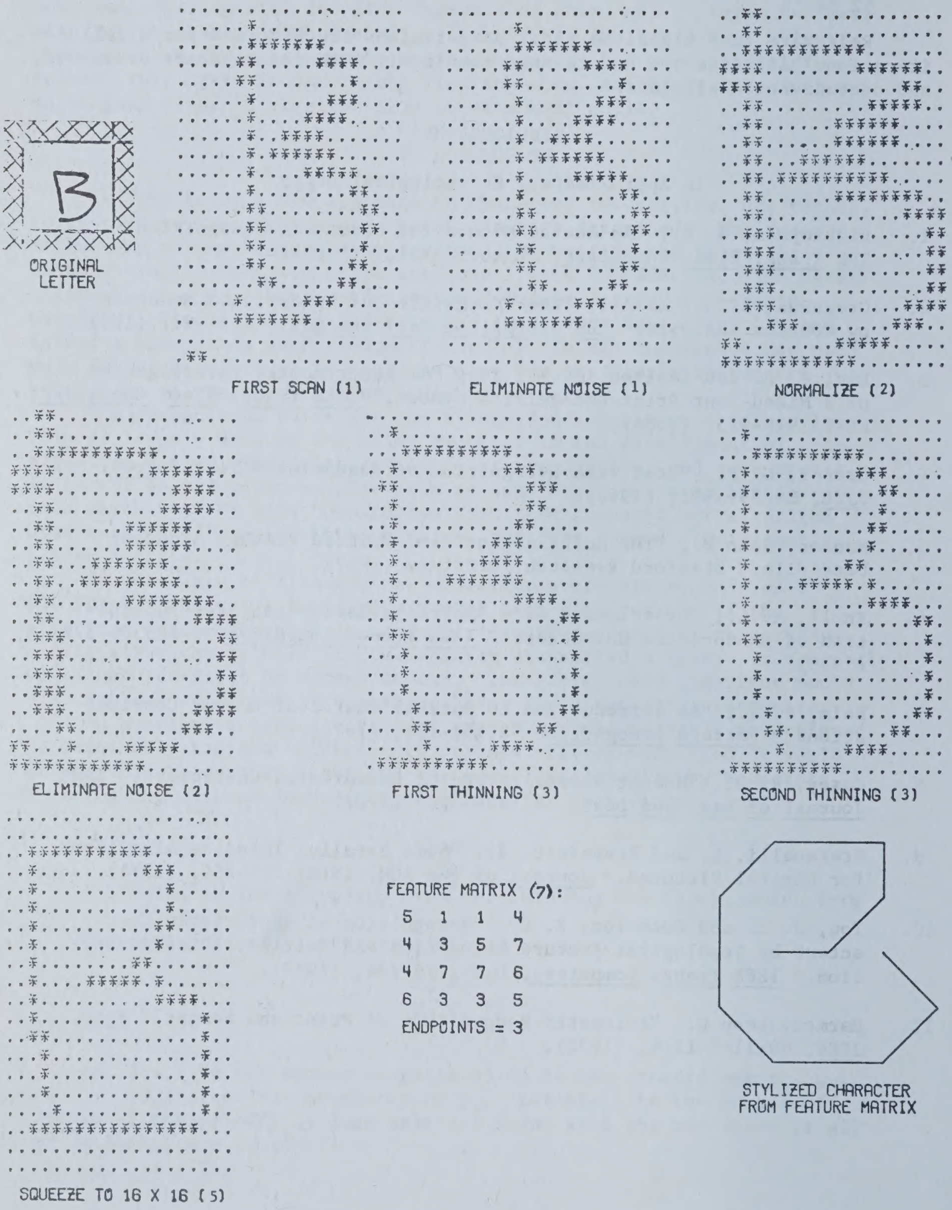

ORIGINAL LETTER
FIRST SCAN (1)
ELIMINATE NOISE (1)
NORMALIZE (2)
ELIMINATE NOISE (2)
FIRST THINNING (3)
SECOND THINNING (3)
SQUEEZE TO 16 X 16 (5)
FEATURE MATRIX (7):
5 1 1 4
4 3 5 7
4 7 7 6
6 3 3 5
ENDPOINTS = 3
STYLIZED CHARACTER FROM FEATURE MATRIX

FIGURE 1–ALGORITHM 3.1 APPLIED TO A TYPICAL B,
ONE OF THE MORE COMPLICATED CHARACTER CLASSES

TEST	AVERAGE RECOGNITION RATE	ORDER OF RECOGNITION RATE BY LETTER	% FIRST OR SECOND CHOICE CORRECT
1. A-Z (entire data base).	61.7%		78.7%
2. 0-9 (entire data base).	73.6%		87.5%
3. A-Z (Using 35 "readable" Characters from each class).	72.1%	P 100% O 100% J 100% T 97.1% L 97.1% J 96.4% C 94.3% Y 91.4% J 91.4% U 91.4% F 85.7% I 82.9% Z 80.0% A 77.1% V 74.3% S 74.3% G 74.3% M 71.4% N 68.6% H 68.6% G 68.6% K 60.0% X 54.3% U 54.3% B 51.4% D 42.9% Q 40.0% E 34.3% W 31.4% R 25.7%	85.8%
4. 0-9 (Using 35 "readable" characters from each class).	85.7%	1 100% 0 100% 3 97.1% 7 94.3% 4 94.3% 5 91.4% 9 85.7% 6 85.7% 2 82.9% 8 25.7%	96.0%
5. A-Z (20x20 correlations of same data as Exp. 3).	58.4%		83.1%
6. 0-9 (20x20 correlations of same data as Exp. 4).	47.7%		70.6%

TABLE 1

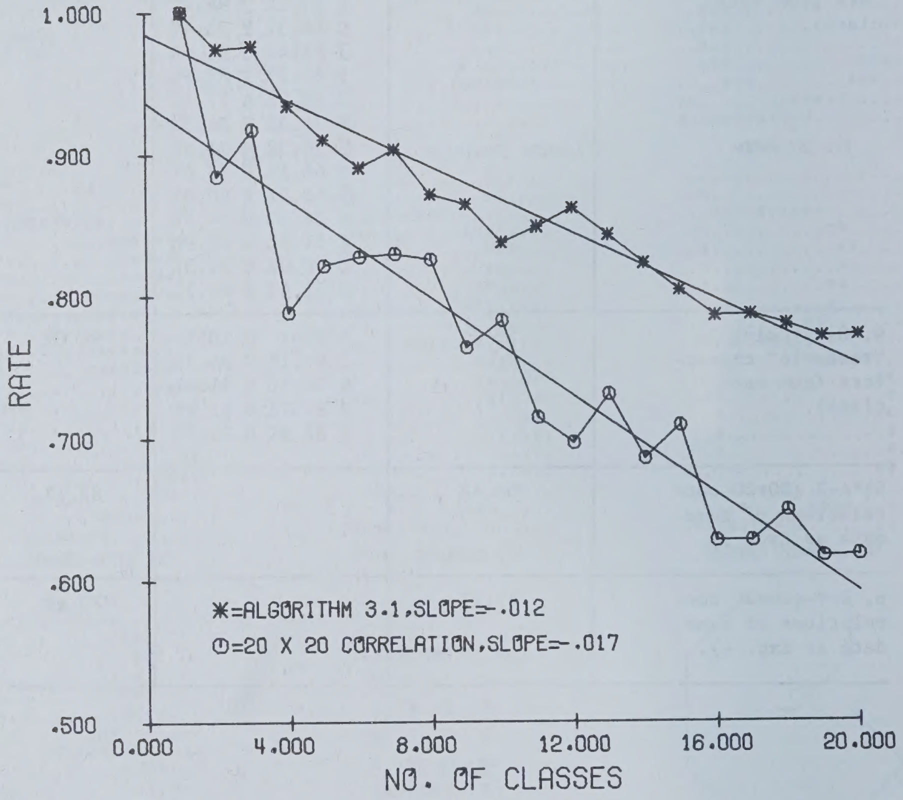


FIGURE 2 - THE AVERAGE RECOGNITION RATES AS A FUNCTION OF THE NUMBER OF CLASSES OF THE EXPERIMENTS.

A SYSTEMS APPROACH TO IDEA RETRIEVAL

KEATS A. PULLEN, JR. AND CHARLES W. HOFFMAN
US Army Ballistic Research Laboratories
Aberdeen Proving Ground, MD 21005

ABSTRACT

Ability to retrieve needed information is a function of such parameters as indexing precision, indexing specificity, tagging criteria, critical mass considerations, as well as Mooer's law and Svenonius' principle. The optimization of a retrieval system to minimize idea half-life under these constraints as a systems and information engineering problem is examined.

INTRODUCTION

Information and idea recovery are major problems in present-day technology, partly because of the exponential increase in volume of ideas being generated, and also because of indexing and searching problems. Mooer's law has an important bearing on the recovery problem as well, since if it is simpler for the individual to avoid making or using a search than it is to evaluate the material recovered by such a search, the individual will simply lay the search material aside.

An efficient systems approach to the retrieval of ideas must of necessity be one offering sufficient alternatives to the user that a rejection based on Mooer's law is the inconvenient path to follow rather than the convenient one. Each query of an efficient system must either produce a sufficient number of clearly useful sources or it must be easily confirmed that a negative response probably was correct. It must also be rather easily verified that most if not all of the relevant material has been identified, and the irrelevant material has not been recovered.

Many experiments have been made on idea recovery, and many developmental "systems" have been assembled and then abandoned. It is the purpose of this paper to attempt to establish guidelines on which development of an efficient recovery system must be based if it is to function on a variety of information "stores". These guidelines must show how a system must grow as the volume of stored information increases, and they must also indicate how to maximize the speed of distribution of useful ideas without cluttering the system unnecessarily with repetitious material.

By nature, an individual first confronting a retrieval system is going to ask it to recover something "indexed" in the system. By "indexed," we of course mean a paper or a document known to be included in the collection, and known to contain an interesting and useful idea. This may be a report or a document authored by the individual, or it may be a document recently received through the system, and thus known to be included. Recovery of the document in question AND other closely related documents will constitute a "success," whereas failure to recover may lead to an adverse decision and an application of Mooer's law to discredit the system.

Admittedly, a test of recovery like that described is in a sense an unfair test, but at the same time, practical queries are remarkably similar to a request for a known idea from a document system as used to test the recovery from the system. As a consequence, the nature of the formation of the query should be carefully analyzed for guidance in establishing the underlying principles governing idea recovery.

EFFECT OF "CRITICAL MASS" AND FILE SIZE

The probability of recovery of an idea from an information bank containing a reference to a subject in question is a function of the number of documents containing information in the technical area in question. To use a phrase from nuclear physics, the document collection has reached "critical mass" when the probability of recovery of an idea from it reaches a value which makes a retrieval effort worthwhile. This critical-mass level is dependent on the number of documents contained which deal with the subject matter in question, and it depends in a very critical way on the manner of indexing of ideas in the documents and the manner of presentation of the indexed ideas in the retrieval system.

As the size of an idea or information bank grows with the addition of new entries, the density of entries must of necessity increase. Svenonius has shown that there is a limit to the number of documents which an individual is willing to examine in detail, and she suggests that this number is roughly ten. That such a limit could be expected to exist is of course self-evident.

Since the density of entries *must* increase as the total number of cataloged elements increases, it is clearly evident that the indexing phrases must at the same time become more precise, so that the desired elements can be separated from the remainder without recovering an excessive amount of non-relevant material. This means that the radius of the recovery hyper-sphere in multi-dimensional information space must become smaller, the more densely the information is packed. The only way this can be accomplished is by an increase of the specificity of the indexing of an idea. Not only must the precision of definition increase as the content in the system is increased, but also the defining level must be adjusted so that it is neither too coarse nor too fine.

BOOLEAN COMBINATION CONSIDERATIONS

Boolean combinations can help in providing the precision required in forming indexing phrases, but actually it is often less efficient than the use of phrases directly. If a set of three words define a specific idea, the total possible number of meanings is a function of the product of the shades of "meanings" for the individual words, i.e.,

$$M_T = \prod_{i=1}^{3} W_i \qquad (1)$$

where W_i is the number of meanings of the word "i," and M_T is the total number of meanings. If, however, these words are combined in a single phrase group, there may be *only one* acceptable technical meaning for the combination. In short, the improvement in definition of the phrase over the Boolean combination of the root words is:

$$M_I = M_P / M_T \qquad (2)$$

where M_P is the number of allowed meanings of the phrase as an aggregate.

As an example, the phrase "solid state physics" has one meaning as a group. The number of definitions for the word components, as listed in Webster's *New World Dictionary* (1957) are:

Solid	18	(total)
State	16	(total)
Physics	4	

The total Boolean combinations of meanings are therefore 1152, of which an unidentified number may not be significant. But in phrase form, a single definition is clearly meant, one related to one of the Boolean combinations, but not defined exactly through it. Clearly, the formation of phrases through Boolean combinations of words is an inefficient way to define an idea.

The phrase must define a single idea, and should define it with precision and at the proper level of specificity. Otherwise, the radius of the recovery hypersphere must be excessively large to recover the desired information, and the relevance ratio will drop significantly. (In addition, Svenonius' principle will apply, and by way of Mooer's law, non-recovery may result.) The only way that recall can be maximized with a minimum of false drops is for both precision and specificity to be optimum.

System documents containing ideas possibly bearing on an idea can be separated into four different groups (forming a recovery matrix):

1. Hits
2. False drops
3. Missed hits
4. Non-relevant elements

If indexing is too broad, there will be many false drops, although there will be few missed documents. If, on the other hand, the technology is defined inprecisely, there will be a substantial population of both false drops and missed hits. If the precision is too high, or the specificity is too high (the phrases define only a small part of the idea to be recovered), then there will be many missed hits. In either case, non-relevant documents will be missed. This has been studied in a recent paper by Kraft[1] and enlarged on by Pullen.[2]

Depending on the depth of understanding of either the system of the subject in question by the user, he or she may have to enter the system through broader concepts, and narrowing the field by addition of modifier ideas. Usually this is accomplished through entering the system through a specific word known to have considerable precision as a definer of the desired idea hypersphere, and refining the search by reviewing the index phrases related to the entry word. If one were searching for some specialized kinds of polynomials, for example, one might enter the system through the word, "Polynomial," and examine the associated phrase defining terms. In the BRL Technical Information Index (TII), some 21 choices are available, with six authors, and five different documents. The phrases are:

Polynomials in several variables
Polynomial integration
Polynomials/discrete orthogonal
Polynomials/Orthogonal (two)
Polynomial/Bessel
Polynomial expressions/fitting
Polynomials/Half-order
Polynomials/nonsymmetric
Polynomial differentiation
Polynomials/Chebychev (three)
Polynomials/Legendre (three)
Polynomials/Trigonometric
Polynomial/Collapse velocity
Polynomials/Hermite (two)
Polynomials/symmetric

A total of 15 apparently different topics are available through the TII, with a preponderence dealing with orthogonal polynomials. It is easily recognized which these are without reference to Boolean techniques being necessary. With the use of single words for definition, both Boolean combinations plus additional verification are required.

LIFETIME AND HALF-LIFE

The real purpose and function of an idea recovery system is to disseminate ideas at the maximum possible rate. In so doing, it encourages technological advance and shortens the period during which a given idea is important. In other words, the purpose of the system is to minimize the useful life of each idea (its lifetime or its half-life). The following factors contribute to minimization of half-life for ideas:

1. Speed of disclosure
2. Precision of disclosure
3. Specificity of disclosure
4. Effectiveness of announcement system
5. Convenience of use of recovery system (Mooer's law)
6. Quality of indexing in depth

The quality of the indexing is vital to minimization of the half-life of ideas. An idea that is not indexed cannot be found, and the fact that it has been noted is effectively lost except for chance recovery. And such an idea must be re-created by another. (E.g., re-invention of the wheel.)

Normally, indexing is limited to ideas expressed in the title and/or abstract for a document, although in medical circles drugs may be referenced as a matter of practice. Actually, indexing should be based on the principles of textbook indexing. A complete index of ideas is prepared, and then ideas are selected from this list in accordance with a set of guidelines which will assure that new and potentially useful concepts, results, and observations will be recoverable. Each indexing phrase must identify a single specific idea and only one idea. And the identified idea must be one of potential value to others.

The promptness of disclosure of a new idea can have an important bearing on the costs of research in a given area. For this reason, the use of statistical techniques to determine when a phrase is important can introduce a substantial lag into growth of technology. Experience shows that when the basic technology has developed to a given state, typically several people may conceive a given idea at nearly the same time. For this reason, speed of dissemination can minimize the amount of duplication of effort and minimize costs.

A system for rapid publication of "new ideas" can clearly be important to our technologists. At the same time, the announcement system must take cognizance of both Mooer's law and Svenonius' principle in a realistic way so that the announcement system will be effective and useful. In short, a carefully planned system taking advantage of information-engineered files in inversions chosen to provide effective options must be developed.

SYSTEM ACCESS OPTIONS

The potential user of a retrieval system comes to it with one of several options of potential interest to him. The more important of these are:

1. Looking for articles on specific subject fields (e.g. FET's).
2. Looking for information on a specific "general" subject.
3. Looking for information on a specialized idea.
4. Browsing for new ideas in a subject field.
5. Looking for a specific paper by a known author.

A recovery system should provide easy and direct recovery from its files based on any of these options. This recovery system may be either automated or manual, but it ideally should be relatively efficient in either a "hard copy" mode or an electronic mode, as it should be adaptable to small, as well as, large libraries or information centers, and it should in addition make available to small libraries an index of material available in nearby large libraries. The kind of retrieval system designed for a library should be cost-effective; that is, its initial cost and cost of operation should be compatible with the costs of operation of both very small and intermediate and large libraries, and specialized libraries as well. At the same time, it must be effective to use so that it will satisfy Mooer's law and Svenonius' principle as well.

A recovery system must not only provide suggested sources of information on a given idea. It must also provide several options for evaluating the potential value of the sources for the purpose at hand. The first level is the step from the component word to a high-definition phrase. Selection of a relevant phrase using the word serving as the point of entry (or the complete phrase, if the user is that familiar with his subject) makes possible up to several orders-of-magnitude reduction in the number of documents which may finally require review.

Once the choice of a high-definition phrase has been used as a "filter" to eliminate all but the most relevant ideas, further filtering, either through use of sets of high-definition phrases or use of abstracts, or both, can lead to a small sample of highly relevant sources of ideas. Either of these additional filter elements may represent the preferred option. The abstract, for example, can be helpful in evaluating the usefulness of a source which deals principally with the idea in question, as the grouping of ideas in an abstract can spell out rather precisely what is discussed. When the desired idea may be dealt with in a peripheral manner, however, it normally is not mentioned in an abstract, and indexing phrases, with the use of upgraded links and roles if required, may be the only means of recovery.

As an example of the comparative use of these techniques, see Figure 1. The use of an abstract and the use of the index in evaluating the detail contents of the document gives a clear indication of the relative "zones of application" of the two approaches. In either case, the use of the word-to-phrase initial entry can lead to the document, and either the abstract or the phrase index used to check the detail contents of the document.

The phrase index is particularly convenient to use when the words of the phrases are columnized. In that form, a complete set of phrases can be perused in a fraction of the time required to read an abstract. Having the phrases in a form amenable to rapid and efficient scanning is of particular importance.

The design of any retrieval system is based on the disciplines of information engineering and communications engineering. These disciplines must be applied in ways which comply with Mooer's law and Svenonius' principle, and at the same time lead to minimum effective half-life for each significant idea.

REFERENCES

1. Kraft, D. H.; "A Decision Theory View of the Information Retrieval Situation: An Operations Approach," JASIS, September-October 1973, p. 368.
2. Pullen, K. A.; "A Decision Theory View of the Information Retrieval Situation: An Operations Approach," Letter to Editor on above, JASIS, January-February 1974, p. 73.

BALLISTIC RESEARCH LABORATORIES

REPORT NO. 1214

KAPullen, Jr./ilm
Aberdeen Proving Ground, MD
August 1963

ON THE RELATION OF PASCAL'S TRIANGLE TO CERTAIN FUNCTIONS

ABSTRACT

Pascal's Triangle is a tabulation of all integers in an arrangement which permits formation of coefficients of $(1 + x)^m$ by a process of summation. The present report shows that the numbers in the triangle are intimately associated with a wide variety of polynomials and functions. A tabulation of coefficients in terms of the numbers of the triangle is included as an appendix.

INDEX LIST

Approximant
Bessel function
Bessel polynomial
Binomial expansion
Chebychev polynomials
Continued fractions
Fibonacci series
Half-order polynomials
Hermite polynomials
Ladder network
Laguerre polynomial
Lattice network
Legendre polynomial
Matrix products
Nonsymmetric polynomials
Pascal triangle
Power series
Recursion relations
Sub-recursion relations
Symmetric polynomials

FIGURE ONE

DUAL CONTROL OF LINEAR DYNAMIC SYSTEMS WITH UNKNOWN PARAMETERS USING BAYESIAN PREPOSTERIOR ANALYSIS

R. K. MEHRA
Division of Engineering and Applied Physics
Harvard University
Cambridge, Massachusetts

ABSTRACT

In this paper, the normal form of Bayesian preposterior analysis is used to derive optimal open-loop dual control for linear dynamic systems with unknown parameters. The computed controls explicitly take into account the aspects of control, learning and experimental design. A suboptimal approach based on input design for parameter estimation is also proposed.

1. INTRODUCTION

Beginning with the work of Feldbaum [1], the dual control problem has been the subject of several studies [2-5, 10, 11]. It has been realized that no analytical solution to the problem is possible and some sort of approximate solution with nearly optimal properties is desired. It is in this spirit that we explore the applicability of the normal form of Bayesian preposterior analysis [7, 12] to the dual control problem. In the context of econometric models with perfect state information, MacRae [4] develops a similar approach form a slightly different viewpoint. We generalize this approach to state vector models with partial state observations and measurement noise, using some results from the recently developed theory of input design [6].

The organization of the paper is as follows. Section 2 discusses a simple model to illustrate the basic procedure. The main results of this paper are presented in section 3. A suboptimal approach based on input design is considered in section 4.

2. DUAL CONTROL FOR AN IMPULSE RESPONSE MODEL

To illustrate the basic approach of this paper, we first consider a scalar impulse response model which can be handled without any approximations. The model is

$$x(t) = \sum_{i=0}^{n-1} \theta_i u(t-i) \tag{1}$$

$$y(t) = x(t) + v(t), \qquad t=1, \ldots, N. \tag{2}$$

where x(t), u(t), y(t) and v(t) (all scalars) are respectively the state, control, output and gaussian measurement noise (zero mean, variance R). The prior distribution of $\theta^T = [\theta_0, \ldots, \theta_{n-1}]$ is normal with mean $\overline{\theta}_o$ and covariance P_o. The loss function whose expected value is to be minimized with respect to $U = [u(-n+1), \ldots, u(N)]$ is

$$J = \sum_{t=1}^{N} q(x(t)-x_d(t))^2 + r(u(t) - u_d(t))^2 \tag{3}$$

As the data set $\{Y_t, U_t\} = \{y(1), \ldots, y(t), u(-n), \ldots, u(t)\}$ becomes available, the mean $\hat{\theta}_t$ and covariance P_t of the posterior density of θ are computed on-line using a Kalman filter

$$\hat{\theta}_t = \hat{\theta}_{t-1} + K_t(y(t) - H(t)\hat{\theta}_{t-1}) \tag{4}$$

$$K_t = P_t H^T(t) R^{-1} \tag{5}$$

$$P_t^{-1} = P_{t-1}^{-1} + H^T(t) R^{-1} H(t) \tag{6}$$

where

$$H(t) = [u(t), \ldots, u(t-n+1)] \tag{7}$$

The information state of the system at time t is represented by $I_{t-1} = \{H(t-1), \hat{\theta}_{t-1}, P_{t-1}\}$ since the posterior distribution $p(y(t+\tau) | Y_{t-1})$ can be obtained from I_{t-1} for $\tau>0$. We now take the expectation in (3) of J with respect to this time-varying information set. Let

$$J_t = q(x(t) - x\ (t))^2 + r(u(t) - u_d(t))^2 \tag{8}$$

Then

$$E\{J_t | Y_{t-1}, U_{t-1}\} = q(H(t)\hat{\theta}_{t-1} - x_d(t))^2 + qH(t)P_{t-1}H^T(t) + r(u(t) - u_d(t))^2 \tag{9}$$

We use the normal form of Bayesian preposterior analysis [7] to compute the optimal openloop U. In this approach, one computes the preposterior mean of J by taking expectations with respect to the prior distribution of Y's. From (9), we obtain

$$E_o(J_t) = q\ E_o\{H(t)\hat{\theta}_{t-1} - x_d(t))^2\} + qH(t)P_{t-1}H^T(t) + r(u(t) - u_d(t))^2 \tag{10}$$

where E_o denotes expectation with respect to the prior distribution of θ viz $p(\theta) \sim N(\overline{\theta}_o, P_o)$. From Eq. (4)

$$E_o(\hat{\theta}_t) = E_o(\hat{\theta}_{t-1}) + K_t H(t)[\overline{\theta}_o - E_o(\hat{\theta}_{t-1})] = \overline{\theta}_o$$

Eq. (10) reduces to

$$E_o(J_t) = q\{(H(t)\overline{\theta}_o - x_d(t))^2 + H(t)(P_o + P_{t-1})H^T(t)\} + r(u(t) - u_d(t))^2 \tag{11}$$

The preposterior mean of J is

$$E_o(J) = \sum_{t=1}^{N} E_o(J_t) \tag{12}$$

which includes the term $J_L = q \sum_{t=1}^{N} H(t)P_{t-1}H^T(t)$ reflecting the effect of learning. If there is no learning; $P_t = P_o$ and the cost $E_o(J)$ will be higher. Notice that all the terms in Eq. (12) except $\underline{J_L}$ are quadratic in U since H(t) is linear in U. Thus for a given P_t sequence $\overline{P}$, the optimal open-loop control $U_o^*(\overline{P})$ based on information upto time zero can be easily computed. The satisfaction of the constraint (6) may be achieved by using Lagrange multiplier matrix $\Lambda(t)$ and defining the Lagrangian function

$$\mathscr{L}(U) = E_o(J) + \mathrm{Tr} \sum_{t=1}^{N} \Lambda_t [P_t^{-1} - P_{t-1}^{-1} - H^T(t)R^{-1}H(t)] \tag{13}$$

Then setting $\dfrac{\partial \mathscr{L}}{\partial u(t)} = 0$, $\dfrac{\partial \mathscr{L}}{P_{t,ij}^{-1}} = 0$ and $\dfrac{\partial}{\partial \Lambda_{t,ij}} = 0$

we obtain a set of linear recursive equations to solve for u(t) in terms of Λ_t which in turn obeys the adjoint set of equations,

$$\Lambda_t = \Lambda_{t+1} + qP_tH^T(t)H(t)P_t \tag{14}$$

$$\Lambda_N = 0$$

$$\sum_{j=t}^{t+n-1} (H(j)\theta_o - x_d(i))c(j)\overline{\theta}_o + H(t)(P_o + P_{t-1})c^T(j) + r(u(t) - u_d(t))$$
$$- \mathrm{Tr}\{\Lambda_t H^T(t)R^{-1}c(j)\} = 0 \tag{15}$$

where c(j) is 1xn vector of all zeros except 1 in jth cloumn. The set of equations (15) is linear in U and can be solved to express U in terms of the sequences P_t and Λ_t. Substituting back in Eq. (6) and (14), one obtains a nonlinear two-point-boundary valye problem in the variables P_t and Λ_t. These equations may be solved via quasilinearization [8] or various other methods [9].

The optimal openloop dual control obtained above may be implemented in a number of different ways. It may be used only for one step and when y(1) becomes available, a new open-loop control may be computed and implemented for the next step and so on. Alternatively, the open loop dual control may be used as a nominal around which the problem amy be linearized as in Tse and Bar-Shalom [10] or the optimal return function approximated quadratically as in Chow [11].

Example:

Consider the case of a scalar unknown parameter θ. Then P_t and Λ_t are scalars and H(t) = u(t). Eq. (15) gives

$$(u(t)\theta_o - x_d(t))\theta_o + u(t)(P_o + P_{t-1}) + r(u(t) - u_d(t))$$
$$- \Lambda_t R^{-1} u(t) = 0$$

or

$$u(t)[\theta_o^2 + P_o + P_{t-1} + r - \Lambda_t R^{-1}] = x_d(t)\theta_o + ru_d(t)$$

$$u(t) = \frac{x_d(t)\theta_o + ru_d(t)}{\theta_o^2 + P_o + P_{t-1} + r - \Lambda_{t/R}} = \frac{c(t)}{d + P_{t-1} - \Lambda_{t/R}}$$

where c(t) and d are known.

$$P_t^{-1} = P_{t-1}^{-1} + u^2(t)/R, \qquad P_o \text{ given}$$

$$\Lambda_t = \Lambda_{t+1} + q\, P_t^2 u^2(t), \qquad \Lambda_N = 0$$

Consider three cases:

1. Certainty Equivalent Control $u_{c_e}(t)$ is obtained by setting P_o, P_t and Λ_t equal to zero.

   $$u_{ce}(t) = \frac{c(t)}{\theta_o^2 + r}$$

2. Control $u_{n\ell}(t)$ for the case of no learning is obtained by setting $P_t = P_o$, $\Lambda_t = 0$

   $$u_{n\ell}(t) = \frac{c(t)}{\theta_o^2 + r + 2P_o} \Rightarrow |u_{n\ell}(t)| \leq |u_{ce}(t)|$$

3. For the dual control u(t), the magnitude of $(P_{t-1} - \Lambda_{t/R})$ determines whether it is smaller or larger than $u_{n\ell}$ e.g. $|u(t)| > |u_{n\ell}(t)|$ if $|P_{t-1} - \Lambda_{t/R}| < P_o$.

Thus the no learning or risk averse control is always less than the certainty equivalent control, but may or may not always be less than the dual control.

3. GENERAL LINEAR STATE VECTOR MODEL

In this section, we apply the general approach of section 2 to the state vector model

$$x(t+1) = \Phi x(t) + Gu(t) + w(t) \tag{16}$$

$$y(t) = Hx(t) + v(t), \quad t=1, \ldots, N \tag{17}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^q$, $w(t) \in \mathbb{R}^n$ is gaussian white noise with zero mean, covariance Q, $y(t) \in \mathbb{R}^p$ and $v(t) \in \mathbb{R}^p$ is also gaussian white noise with zero mean and covariance R. Let $\theta \in \mathbb{R}^m$ be the vector of unknown, random parameter from Φ, G, Q, and R. The prior distribution of θ is assumed normal with mean θ_o and covariance M_o^{-1}. The prior distribution of x(o) is also normal with mean $\bar{x}_o$ and covariance P_o. The random variables x(o), w(t), v(t) and θ are assumed independent from each other. The loss function is

$$J = \frac{1}{2} x^T(N)S_N x(N) + \frac{1}{2} \sum_{t=0}^{N-1} \{x^T(t)Ax(t) + u^T(t)Bu(t)\} = \sum_{t=0}^{N-1} J_t \quad (18)$$

where J_t is easily defined by correspondence.

To solve this problem by our approach, it will be necessary to make certain simplifying assumptions. The first assumption is

A1. The parameter estimates $\hat{\theta}_t$ based on $\{Y_t, U_t\}$ are unbiased and efficient i.e.

$$E[\hat{\theta}_t] = \theta$$

$$\mathrm{Cov}[\hat{\theta}_t] = M_t^{-1}$$

where M_t is the Generalized Fisher Information Matrix based on $\{Y_t, U_t\}$ and the prior distribution of θ. An expression for M_t was derived in Ref. [6] and will be given below. We further assume that the total sample size equivalent to the given prior information and the data is large so that a generalized maximum likelihood estimate of θ has the above properties and is furthermore normally distributed [12]. M_t is defined as

$$M_t = \underset{\theta, Y_t}{E} \{\frac{\partial}{\partial\theta} \log p(Y_t, \theta)(\frac{\partial}{\partial\theta} \log p(Y_t, \theta))^T\} \quad (19)$$

where $p(Y_t, \theta) = p(Y_t|\theta)p(\theta)$. Here $p(\theta)$ is the prior density on θ and $L(\theta) = \log p(Y_t|\theta)$ is obtained by using a Kalman state estimator as

$$L(\theta) = -\frac{1}{2} \sum_{t=1}^{N} \log|\Sigma(t)| - \frac{1}{2} \sum_{t=1}^{N} \nu^T(t)\Sigma^{-1}(t)\nu(t)$$

where $\nu(t)$ is the innovation sequence

$$\nu(t) = (y(t) - H(\theta)\hat{x}(t,\theta)), \quad \Sigma(t) = HP(t,\theta)H^T + R \quad (21)$$

$$\hat{x}(t,\theta) = E[x(t)|Y_{t-1}, \theta], \quad P(t,\theta) = \mathrm{Cov}(\hat{x}(t,\theta)) \quad (22)$$

$\Sigma(t)$ is the covariance of $\nu(t)$ obtained from a Kalman Filter.

From Eq. (19)

$$M_t = E\{\underset{\theta}{E}\, \underset{Y_t|\theta}{}\, \frac{\partial}{\partial\theta} \log p(Y_t|\theta))^T\} + M_o \quad (23)$$

The computation of the first term is rather involved and is given in Ref. [6]. We only state the final results have

$$M_t \approx M_t'(\theta_o) + M_o \quad (24)$$

In Eq. (24), a term of the type $\frac{1}{2} \mathrm{Tr}[\frac{\partial^2 M'}{\partial\theta^2}|_{\theta_o} M_o^{-1}]$ has been dropped assuming that it is much smaller than the above two terms. An expression for $M_t'(\theta_o)$ is obtained in terms of the mean and covariance of the augmented state vector $\hat{x}_A(t)$ consisting of $\hat{x}(t)$ and the columns of $\left.\frac{\partial \hat{x}(t)}{\partial \theta}\right|_{\theta_o}$ viz. $\frac{\partial \hat{x}(t)}{\partial \theta_1}, \ldots, \frac{\partial \hat{x}(t)}{\partial \theta_m}$. The (i,j)th element of M_t is given by

$$M_t(i,j) = M_{t-1}(i,j) + \mathrm{Tr}\{c_j[\bar{x}_A(t)\bar{x}_A^T(t) + \textstyle\sum_A(t)]c_i^T H^T \textstyle\sum^{-1}(t) H$$

$$+ \frac{1}{2} \textstyle\sum^{-1}(t) \frac{\partial \sum(t)}{\partial \theta_i} \textstyle\sum^{-1}(t) \frac{\partial \sum(t)}{\partial \theta_j}\}$$

$$+ \frac{1}{4} \mathrm{Tr}(\textstyle\sum^{-1}(t) \frac{\partial \sum(t)}{\partial \theta_i}) \mathrm{Tr}(\textstyle\sum^{-1}(t) \frac{\partial \sum(t)}{\partial \theta_j}) \tag{25}$$

where $c_i = [0, 0, \ldots, I, 0, \ldots, 0]$ and $\bar{x}_A(t)$, $\sum_A(t)$ and $\frac{\partial \sum(t)}{\partial \theta_i}$ obey recursive equations. The only equation involving u(t) is

$$\bar{x}_A(t+1) = \Phi_A \bar{x}_A(t) + G_a u(t) \tag{26}$$

See Ref. [6] for defintions of Φ_A and G_A. We now compute the posterior mean of J_t conditional on Y_{t-1} and θ.

$$E[J_t|Y_{t-1},\theta] = \frac{1}{2}\{\hat{x}^T(t,\theta)A\hat{x}(t,\theta) + \mathrm{Tr}[AP(t,\theta)] + u^T(t)Bu(t)\} \tag{27}$$

Further expectation with respect to the posterior of θ assuming a first order expansion of $\hat{x}(t,\theta)$ around $\hat{\theta}_t$ provides

$$E[J_t|Y_{t-1}] = \frac{1}{2}\{\hat{x}^T(t,\hat{\theta}_t)A\hat{x}(t,\hat{\theta}_t) + u^T(t)Bu(t)$$

$$+ \mathrm{Tr}[A(M_t^{-1} + \left.\frac{\partial \hat{x}(t)}{\partial \theta}\right|_{\hat{\theta}_t} M_t^{-1} \left.\frac{\partial \hat{x}^T(t)}{\partial \theta}\right|_{\hat{\theta}_t})]\}$$

By using the augmented state vector $\hat{x}_A$, this equation may be written as

$$E[J_t|Y_{t-1}] = \frac{1}{2}\{\hat{x}(t,\hat{\theta}_t) \mathscr{A} \hat{x}_A(t,\hat{\theta}_t) + u^T Bu(t) + \mathrm{Tr}(AM_t^{-1})\}$$

Taking expectations with respect to Y_t, the preposterior mean is

$$E_o(J) = \sum_{t=0}^{N} E_o[J_t] = \frac{1}{2} \sum_{t=0}^{N} \{\bar{x}_A^T(t,\theta_o) \mathscr{A}(t)\bar{x}_A(t,\theta_o) + u^T(t)Bu(t)$$

$$+ \mathrm{Tr}[\mathscr{A}(t)\textstyle\sum_A(t,\theta_o) + AM_t^{-1}]\} - \frac{1}{2} u^T(N)Bu(N) \tag{29}$$

where $\sum_A(t)$ is given by [6]

$$\textstyle\sum_A(t+1) = \Phi_A \sum_A(t)\Phi_A^T + K_A K_A^T \tag{30}$$

K_A consists of Kalman gain and its partials with respect to θ.

To obtain the optimal open loop dual control $u^*(t)$, we minimize (29) subject to the constraints (25)-(26). The remaining equations for $\Sigma_A(t)$, $\Sigma(t)$ and $\frac{\partial \Sigma(t)}{\partial \theta_i}$ need not be adjoined to the performance index since they do not depend on $u(t)$. To simplify notation, we rewrite (25) as

$$M_t = M_{t-1} + [\bar{x}_A^T(t) D_{ij}(t) \bar{x}_A(t) + E_{ij}(t)]_{\substack{i=1,\ldots,m \\ j=1,\ldots,m}} \tag{31}$$

where $D_{ij}(t)$ and E_{ij} are defined by correspondence. The first order necessary conditions of optimality are easily stated in terms of the Hamiltonian

$$\mathcal{H}(t) = \frac{1}{2} \bar{x}_A^T(t) \mathcal{A}(t) \bar{x}_A(t) + u^T(t) B u(t) + \mathrm{Tr}[\mathcal{A}(t) \Sigma_A(t) + A M_t^{-1}]$$

$$+ \lambda^T(t+1)[\Phi_A \bar{x}_A(t) + G_A u(t)] + \mathrm{Tr}\{\Lambda^T(t+1)(M_t + [\bar{x}_A^T(t) D_{ij}(t) \bar{x}_A(t)$$

$$+ E_{ij}(t)])\} \tag{32}$$

where $\lambda(t)(n \times 1)$ are the Lagrange multiplier functions for Eq. (26) and $\Lambda(t)(m \times m)$ are the Lagrange multiplier functions for Eq. (31).

The first order necessary conditions of optimality are Eqs. (25)-(26) along with

$$\frac{\partial \mathcal{H}(t)}{\partial u(t)} = 0 \tag{33}$$

$$\lambda(t) = -\left(\frac{\partial \mathcal{H}(t)}{\partial x(t)}\right)^T, \qquad \lambda(N) = 0 \tag{34}$$

$$\Lambda(t) = -\frac{\partial \mathcal{H}(t)}{\partial M_t(t)}, \qquad \Lambda(N) = 0 \tag{35}$$

One method of solving these equations is Conjugate Gradient Techniques. [15] We assume a nominal $u(t)$ history and solve Eqs. (25)-(26) forward in time. Then Eqs. (34) and (35) are solved backward in time and $\frac{\partial \mathcal{H}}{\partial u(t)}$, the gradient function, is computed simultaneously. If the gradient is nonzero, $u(t)$ is updated in a conjugate direction that results in a decrease in the cost function $E_o(J)$. This process is repeated till convergence to a local minimum is achieved.

Remark: 1. The sensitivity function calculations can be simplified by using results of Wilkie and Perkins [12] and Gupta and Mehra [13].
2. Optimal feedback control strategies of known form such as linear state variable feedback etc. can also be computed using $E_o(J)$.

4. A SUBOPTIMAL APPROACH

As can be seen from section 3, the optimal solution for dual control is quite involved and requires the solution of a two-point boundary value problem. Moreover, the solutions obtained are only locally optimal. In Ref. [6], a globally optimal solution was given for the problem of minimizing $\mathrm{tr}(M_N^{-1})$ subject to the constraint

$$E_o\left[\sum_{t=1}^{N} x^T(t)Ax(t) + u^T(t-1)Bu(t-1)\right] \leq c$$

where c is a given constant. The optimal control is randomized and is given in terms of the input autocorrelation function R_{uu} (NqxNq) with elements $R_{uu}(\tau,s) = E[u(\tau)u^T(s)]$ or a transformed version

$$R_{u'u'} = \frac{1}{c'} D^{1/2} R_{uu} (D^{1/2})^T$$

where c' and D are specified in terms of the known system parameters. A minmax property is used to compute and verify the optimal input autocorrelation function $\hat{R}_{u'u'}$. For details, see Ref. [6].

The above formulation gives an input that accounts for the experimental design aspects of dual control, but does not include learning aspects during the N periods. It would be particularly suitable for the case in which the parameters are updated every N periods. For dual control purposes, it may be more suitable to solve the mathematical dual problem of $\min_u E_o[\sum_{t=1}^{N} x^T(t)Ax(t) + u^T(t-1)Bu(t-1)\}$ subject to the constraint $\mathrm{tr}(M_N^{-1}) \leq d$.

5. CONCLUSIONS

It is shown how open loop dual control strategies having features of learning and experimental design may be computed for linear dynamic systems with partial noisy state observations.

REFERENCES

[1] A. A. Feldbaum, "Dual Control Theory Parts I-IV" Automation and Remote Control, Vol. 21, 1960, No. 9; pp. 874-880, No. 11, pp. 1033-1039, Vol. 22, 1961, No. 1, pp. 1-21, No. 2, pp. 109-121.

[2] T. Bohlin, "Optimal Dual Control of a Simple Process with Unknown Gain" IBM Nordic Lab. TP18.196, 1969.

[3] J. J. Åström and B. Wittenmark, "Problems of Identification and Control" J. of Math. Anal. and Appl. Vol. 34, 1971, pp. 90-113.

[4] E. C. MacRae, "Linear Decision with Experimentation" Annals of Economic and Social Measurement, 1/4, 1972.

[5] G. C. Rausser and J. W. Freebairn, "Approximate Adaptive Control Solutions to U.S. Beef Trade Policy" Annals of Economic and Social Measurement, Vol. 3, No. 1, January 1974.

[6] R. K. Mehra, "Optimal Input Signals for Parameter Estimation in Dynamic Systems -- Survey and New Results" IEEE T-AC, December 1974.

[7] H. Raiffa and R. Schlaifer, Applied Statistical Decision Theory, Division of Research, Graduate School of Business Administration, Harvard University, Boston, 1961.

[8] R. W. Bellman and R. E. Kalaba, Quasilinearization and Nonlinear Boundary Value Problems, American Elsevier, New York, 1965.

[9] A. E. Bryson and Y. C. Ho, Applied Optimal Control, Xerox College Publishing, 1969.

[10] E. Tse and Y. Bar-Shalom, "An Activity Adaptive Control for Linear Systems with Random Parameters via the Dual Control Approach" IEEE T-AC, Vol. 18, No. 2, 1973.

[11] G. Chow, "A Solution to Optimal Control of Linear Systems with Unknown Parameters" Third NBER Stochastic Control Conference, Washington, D. C. 1974.

[12] M. DeGroot, Optimal Statistical Decisions, McGraw Hill, 1970.

[13] D. F. Wilkie and W. R. Perkins, "Generation of Sensitivity Functions for Linear Systems Using Low Order Models" IEEE T-AC, April 1969.

[14] N. K. Gupta and R. K. Mehra, "Computational Aspects of Maximum Likelihood Estimation and Reduction in Sensitivity Function Calculations" IEEE T-AC, December 1974.

[15] R. K. Mehra and A. E. Bryson, "Conjugate Gradient Methods with an Application V/STOL Flight Path Optimization" AIAA J. of Aircraft, Vol. 6, No. 2, March-April 1969.

ACKNOWLEDGMENT

The research reported in this paper was made possible through support extended by the Division of Engineering and Applied Physics, Harvard University, and the U. S. Office of Naval Research under the Joint Services Electronics Program by Contract N00014-67-A-0298-0006.

OBSERVERS AND PSEUDO OBSERVERS FOR LINEAR TIME DELAY SYSTEMS

CAPT R. V. GRESSANG
Air Force Flight Dynamics Laboratory
Wright-Patterson AFB, Ohio

ABSTRACT

Observers for linear time delay systems and their qualitative properties are studied. Sufficient conditions for existence are presented and compared with previous results. Pseudo observers, which are algebraically synthesizable systems, are defined and a sufficient condition for existence given. Pseudo observer properties are shown to be similar to observer properties.

I. INTRODUCTION

In 1964, Luenberger [1] introduced a method for approximating the state of a finite dimensional linear system given incomplete measurements. The method consists of constructing a second linear system, called an observer. The conditions for an observer to exist are closely related to observability of the original system, and observer properties make them useful in feedback control systems.

Both Lee [2] and Hewer and Nazaroff [3] have partially extended observer theory to time delay systems. Lee considered identity observers, in conjunction with system identification procedures. Hewer and Nazaroff considered reduced order observers (observers which do not reconstruct available measurements). Their directly useful results for observer synthesis are for systems with small time delays.

This paper provides a method to reduce identity observer design for a class of time delay systems to a problem equivalent to pole placement using state feedback, improving certain results of Lee and of Hewer and Nazaroff. Hewer and Nazaroff's result on observers in linear feedback control systems is extended to arbitrary linear control laws, such as arise from quadratic loss optimal control problems.

A system called a pseudo observer is defined, and the existence proof for pseudo observers furnishes an algebraic synthesis procedure. Pseudo observers are shown to be usable in closed loop control system design, if a modified set of system equations is used for control law design.

Section II defines an observer for a time delay system and discusses the observer's general properties. Also in this section, the spectral separation property possessed by finite dimensional closed loop control systems using observers is extended to time delay systems. Section III treats sufficient conditions for the existence of identity and reduced order observers. Section IV introduces and discusses pseudo observers, and Section V states the conclusions.

II. PRELIMINARIES

This paper considers linear time invariant differential delay equations

$$x(t) = \sum_{i=0}^{k} A_i\, x(t-h_i) + \sum_{i=0}^{k} B_i u(t-h_i) \tag{1}$$

where $0 = h_o \leq h_1 \leq --- \leq h_k < \infty$, and the initial functions are in $C_n[-h_k,0]$. The observations are given by

$$y(t) = C\, x(t). \tag{2}$$

Equation (1) has a unique differentiable solution if u(t) is continuous. If u(t) satisfies a Caratheodory condition, equation (1) is satisfied almost everywhere by a unique absolutely continuous function. [4]

An observer for the system of equations (1) and (2) is defined as

$$z(t) = \sum_{i=0}^{k} F_i z(t-h_i) + \sum_{i=0}^{k} G_i y(t-h_i) + \sum_{i=0}^{k} H_i u(t-h_i) \tag{3}$$

with initial functions in $C_r[-h_k,o]$. Equation (3) must satisfy the following three conditions:

1. There exists an r x n matrix T such that

$$TA_i - F_i T = G_i C \text{ for } i=0,1, ---, k. \tag{4}$$

2. There exist n x m and n x r matrices M and N such that

$$MC + NT = I. \tag{5}$$

3. For i = 0, 1, ---, k, the matrices H_i satisfy

$$T\, H_i = B_i. \tag{6}$$

Here n is the dimension of x(t), r is the dimension of z(t), and m is the dimension of y(t).

If r = n, and T = I, equation (3) is an identity observer. If r = n-m, and M and N are non-zero, then equation (3) is a reduced order observer. Note that equations (4), (5), and (6) are independent of the magnitudes of the delays. Define the observer error by

$$e(t) = z(t) - T\, x(t). \tag{7}$$

Then exactly parallel to Luenberger's development, [1] the following lemma can be proved.

Lemma 1: The observer error e(t) satisfies the homogeneous linear differential delay equation

$$e(t) = \sum_{i=0}^{k} F_i e(t-h_i) \tag{8}$$

where $e(t) = z(t) - T\, x(t)$ for $t\varepsilon[-h_k,0]$.

Because of Lemma 1, the control u(t) can be ignored in designing observers, and only homogeneous systems considered. For a practical observer, the observer error must decay to zero, thus equation (8) must be asymptotically stable. Observer design reduces to satisfying equation (4) while requiring equation (8) to be asymptotically stable. For an identity observer asymptotic stability is equivalent to finding matrices G_i so that u(t) = y(t) is a stabilizing control for

$$s(t) = \sum_{i=0}^{k} A_i\, x(t-h_i) + \sum_{i=0}^{k} G_i u(t-h_i). \tag{9}$$

Equation (8) is asymptotically stable if all roots of

$$\mathrm{Det}\left[Is - \sum_{i=0}^{k} F_i e^{-sh_i} \right] = 0 \tag{10}$$

have negative real parts. [4] At most a finite number of roots of equation (10) have non-negative real parts; however these roots are difficult to determine. Routh-Hurwitz type tests for all roots to lie in the left half plane are complicated [5], so applying the condition that all roots of equation (10) have negative real parts is difficult.

Theorem 2 and Corollary 1 of Hewer and Nazaroff [3] together imply use of the conjecture that if all the eigenvalues of F_o and F_i lie in the left half plane, then all the roots of

$$\mathrm{Det}\left[Is - F_o - F_e^{-sh_1} \right] = 0 \tag{11}$$

lie in the left half plane.

However, the system

$$\dot{x}(t) = -1.0x(t) - 5.0x(t-1) \tag{12}$$

furnishes a counter example to this conjecture. A necessary and sufficient condition for all roots of

$$s + 1.0 + 5.oe^{-s} = 0 \tag{13}$$

to lie in the left half plane is [5]

1. $-1.0 < 0$
2. $-1.0 < 5.0 < \sqrt{a_1^2 + 1}$

where a_1 is the root of $a = -\tan a$ such that $0 < a < \pi$. But $\sqrt{a_1^2 + 1} \overset{\sim}{=} 2.24$, and condition 2 is violated. Equation (13) thus has right half plane or imaginary axis zeros; so equation (12) is not asymptotically stable.

In Equation (1), if only the B_0 control term is non-zero, Ross [6] showed that the control solving a quadratic optimal control problem is a linear functional of the state. The feedback control form is

$$u(t) = \int_{-h_k}^{o} [dk(\theta)]\ x\ (t+\theta) \tag{14}$$

where $k(\theta)$ is a matrix of functions of bounded variation.

Apply a feedback control of the form of equation (14) to equations (1) and (2). Since $x(t)$ cannot be measured directly,

$$\hat{x}(t) = M\ y(t) + N\ z(t) \tag{15}$$

replaces $x(t)$ in the control law. $z(t)$ is the output of an observer.

Changing variables from $x(t)$ and $z(t)$ to $x(t)$ and $e(t)$, and using the observer relations [7], equations (1), (2), (3), (14) and (15) become

$$\begin{aligned} x(t) = {} & \sum_{i=0}^{k} A_i x(t-h_i) + \sum_{i=0}^{k} B_i \int_{-h_k}^{o} [dk(\theta)] x(t-h_i+\theta) \\ & - \sum_{i=0}^{k} B_i \int_{-h_k}^{o} [dk(\theta)] N\ e(t-h_i+\theta) \end{aligned} \tag{16}$$

$$e(t) = \sum_{i=0}^{k} F_i e(t-h_i) \tag{17}$$

A "pole-separation" property identical to that for finite dimensional systems can be proved using equations (16) and (17).

Lemma 2: The spectrum of the closed loop system formed by a state feedback control in equation (1) in conjunction with an observer is the union of the spectrum of equation (1) driven by state feedback control with the spectrum of the observer.

Proof: The spectrum of the closed loop system is the roots of

$$\text{Det} \left[\begin{array}{c|c} Is - \sum_{i=0}^{k} A_i e^{-sh_i} - \sum_{i=0}^{k} B_i \int_{-h_j}^{o} [dk(\theta)] e^{s\theta} e^{-sh_i} & \sum_{i=0}^{k} B_i \int_{-h_j}^{o} [dk(\theta)] N e^{s\theta} e^{-sh_i} \\ \hline 0 & Is - \sum_{i=0}^{k} F_i e^{-sh_i} \end{array} \right] = 0 \tag{18}$$

Using properties of determinants, equation (18) is rewritten as

$$\mathrm{Det}\left[Is - \sum_{i=0}^{k} A_i e^{-sh_i} - \sum_{i=0}^{k} B_i \int_{-h_k}^{o} [dk(\theta)] e^{s\theta} e^{-sh_i} \right]$$

$$\cdot\ \mathrm{Det}\left[Is - \sum_{i=0}^{k} F_i e^{-sh_i} \right] = 0 \qquad (19)$$

The first term of equation (19) yields the spectrum of equation (1) with state feedback control, and the second term determines the observer spectrum. Equation (19) has a root if the first, second, or both terms have a root. Q.E.D.

Lemma 2 can also be established as a special case of a general result derived using spectral theory in Gressang. [8] Lemma 2 allows time delay control and observer synthesis to be decoupled in a way similar to the decoupling possible in finite dimensional systems.

III. IDENTITY AND REDUCED ORDER OBSERVERS

This section presents sufficient conditions for constructing identity and reduced order observers. The identity observer result requires fewer hypotheses than Lee's result, [2] and the proof is entirely different. It reduces the stability requirement to the equivalent of a pole placement problem for a finite dimensional system. The identity observer result can be converted to a stabilization result for time delay systems. The reduced orber observer result is the conversion of a stabilization result of Ross. [6] The reduced order observer produced has no delays in its unforced dynamics.

The identity observer result is as follows:

Theorem 1: An identity observer exists for equations (1) and (2) if the pair of matrices A_o, C is observable. The identity observer has the form

$$\dot{z}(t) = [A_o + GC]z(t) + \sum_{i=1}^{k} A_i z(t-h_i) - Gy(t). \qquad (20)$$

Proof: From equations (7) and (8) the theorem is proved if

$$\dot{e}(t) = (A_o + GC)\, e(t) + \sum_{i=1}^{k} A_i e(t-h_i) \qquad (21)$$

is asymptotically stable. Thus proof is reduced to selecting G such that equation (21) is asymptotically stable.

A theorem of Razumichin [9] is used to prove G exists. This theorem states that equation (21) is asymptotically stable if a positive definite Liapunov function V(x) exists whose total derivative V(x) is negative definite for all solutions of equation (21) satisfying

$$V(x(s)) \le V(x(t)),\ 0 \le s \le t. \qquad (22)$$

Pick $V(z) = z'(t)\, z(t)$. Condition (22) implies

$$||z(t-h_i|| < ||z(t)||, \ i-1, \ ---, \ k. \tag{23}$$

The total derivative is given by

$$V(z) = z'(t)\ [(A_o+GC) + (A_o+Gc)']\ z(t) + \sum_{i=1}^{k} [z'(t-h_i)\ A_1'\ z(t) + z'(t)\ A_i z(t-h_i)]. \tag{24}$$

Using condition (23) and the triangle inequality in equation (24), and rewriting it as a quadratic form,

$$\dot{V}(z) \leq z'\ (t)\ |(A_o + \sum_{i=1}^{k} 2||A_i||I+GC) + (A_o + \sum_{i=1}^{k} 2||A_1||I+GC)'|\ z(t)$$

$$= z'(t)\ \underline{\overline{X}}\ x(t). \tag{25}$$

Elementary operations leave the rank of a matrix invariant, so

$$C \text{ and } A_o + \sum_{i=1}^{k} 2||A_i||I$$

are observable if C and A_o are observable. Therefore G exists such that

$$A_o + \sum_{i=1}^{k} 2||A_i||I + GC$$

has all its eigenvalues in the left half plane. [10] For this G, the Liapunov matrix equation associated with equation (25) shows that $\underline{\overline{X}}$ is negative definite. [11] Therefore V(z) is negative definite subject to condition (23). Q.E.D.

The reduced order observer condition is given in the following theorem. Hewer and Nazaroff [3] treat a similar case, but do not note that the error equation has no delay terms, nor show how to choose the F_i.

Theorem 2: A reduced order observer exists for the system of equations (1) and (2) if the pair of matrices A_o, C is observable, and the rows of A_i are in the row space of C for $i \geq 1$. The error equation does not contain time delays.

Proof: Since A_o, C form an observable pair, finite dimensional observer theory yields T, F, G_o, M, and N so that

$$TA_o - FT = G_oC \quad (26) \qquad\qquad MC + NT = I \tag{27}$$

and F has only left half plane eigenvalues.

Now consider the k - 1 equations

$$TA_i - F_iT = G_iC, \ i=1, \ ---, \ k. \tag{28}$$

Setting $F_i = 0$ for $i \geq 1$,

$$TA_i = G_iC, \ i - 1, \ ---, \ k. \tag{29}$$

Each row of TA_i is a linear combination of rows of A_i, and the rows of A_i are in the row space of C. Thus each row of TA_i is in the row space of C, and is a linear combination of the rows of C. The coefficients of these linear combinations determine G_i, and solve equation (29) for each $i \geq 1$.

Simultaneous solution of equations (26), (27) and (29) determines the observer, as

$$z(t) = Fz(t) + \sum_{i=0}^{k} G_i y(t-h_i),\ z(o) = 0 \quad (30) \qquad \hat{x}(t) = My(t) + Nz(t) \quad (31)$$

The error differential equation

$$e(t) = F\ e(t) \tag{32}$$

contains no delay terms. Q.E.D.

Theorem 2 may be formally applied to develop reduced order observers for scalar differential-delay equations of retarded type. For simplicity, consider the scalar equation of retarded type with one delay.

$$X^{(n)}(t) + A_{on-1}\ X^{(n-1)}(t) + \text{---}\ A_{oo}X(t) + A_{in-1}X^{(n-1)}(t-h)$$

$$+ \text{---} + A_{io}X(t-h) = b_{n-1}U^{(n-1)}(t)\ \text{---} + b_oU(t). \tag{33}$$

Using a well known technique for reducing scalar ordinary differential equations to vector - matrix form, [12] equation (33) is formally equivalent to

$$x(t) = \begin{bmatrix} -A_{on-1} & 1 & 0 & 0 \\ -A_{on-2} & & & \\ \cdot & & & 0 \\ \cdot & & & \\ \cdot & & & 1 \\ -A_{oo} & & 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} -A_{in-1} & 0 & 0 \\ -A_{in-2} & & \\ \cdot & & \\ \cdot & & \\ -A_{1o} & 0 & 0 \end{bmatrix} x(t-h) + \begin{bmatrix} b_n-1 \\ \cdot \\ \cdot \\ \cdot \\ b_o \end{bmatrix} u(t)$$

$$= A_o\ x(t) + A_1\ x(t-h) + B\ u(t) \tag{34}$$

$$y(t) = [1\ 0\ \text{---}\ 0]\ x\ (t). \tag{35}$$

As the A_o and C matrices of equations (34) and (35) are an observable pair, and the rows of A_1 are in the row space of C, Theorem 2 applies.

IV. PSEUDO OBSERVERS

The observers of the previous section were restricted to special classes of differential delay systems. To widen the classes of applicable systems, the transformation

$$z(t) = T\ x(t) \tag{36}$$

will be generalized to

$$z(t) = \sum_{i=0}^{k} T_i\ x(t-h_i). \tag{37}$$

Using transformations of the type of equation (37), more systems can be treated. However, this transformation does not always allow the state of equation (1) to be estimated stably. Sometimes only the result of a scalar difference operator operating on the components of $x(t)$ can be stably reconstructed, hence systems of this type are called pseudo observers. The precise definition of a pseudo observer is given below.

An algebraic method of treating equations (1) and (2) is now introduced. This allows pseudo observers to be derived by algebraic procedures. Assume that the delays h_i are all integer multiples of some basic delay h, i.e., $h_i = l_i h$. Equations (1) and (2) are rewritten as

$$\dot{x}(t) = A(\Delta)\, x(t) + B(\Delta)\, u(t) \quad (38) \qquad y(t) = C\, x(t) \quad (39)$$

where

$$A(\Delta) = \sum_{i=0}^{k} A_i \Delta^{l_i} \quad (40) \qquad B(\Delta) = \sum_{i=0}^{k} B_i \Delta^{l_i} \quad (41)$$

and Δ represents a delay of h units, i.e.,

$$\Delta x(t) = x(t-h). \tag{42}$$

C could be replaced by $C(\Delta)$ and measurement delays considered without additional complication.

The elements of $A(\Delta)$ and $B(\Delta)$ are polynomials in the indeterminant Δ, and can be embedded in the algebraic field of rational forms in Δ. This allows $A(\Delta)$, $B(\Delta)$ and C to be manipulated using linear algebra, provided the field associated with the coordinate vectors is the field of rational forms in Δ. If end results can be interpreted as differential delay equations, validity can be established by substitution, and formality of intermediate steps is not important. A similar device has been used without a justification by Tzafestas and Parasevopoulos [13] to treat decoupling of time delay systems. For equations (38) and (39), the asymptotically stable equation

$$z(t) = F(\Delta)x(t) + G(\Delta)y(t) + H(\Delta)u(t) \tag{43}$$

is an observer if the elements of $F(\Delta)$, $G(\Delta)$, $H(\Delta)$, $T(\Delta)$, $M(\Delta)$, and $N(\Delta)$ are all polynomials in Δ, and these equations are satisfied:

$$T(\Delta)\, A(\Delta) - F(\Delta)\, T(\Delta) = G(\Delta)\, C(\Delta) \tag{44}$$

$$H(\Delta) = T(\Delta)\, B(\Delta) \quad (45) \qquad M(\Delta)\, C(\Delta) + N(\Delta)\, T(\Delta) = I. \quad (46)$$

If equation (46) is replaced by

$$M(\Delta)\, C(\Delta) + N(\Delta)\, T(\Delta) = q(\Delta)I, \tag{47}$$

where $q(\Delta)$ is a scalar polynomial in Δ, equation (43) defines a pseudo observer.

Define a matrix $W(\Delta)$ by

$$W(\Delta) = [C(\Delta)\;\; C(\Delta)A(\Delta)\;\; \text{---}\;\; C(\Delta)A^{n-1}(\Delta)]' \tag{48}$$

A sufficient condition for existence of a pseudo observer is that $W(\Delta)$ have rank n. The rank condition on $W(\Delta)$ is suggestive of the necessary and sufficient condition for reconstructability of a finite dimensional system. [14]

Theorem 3: If $W(\Delta)$ has rank n, a pseudo observer can be constructed for equations (38) and (39).

Proof: Since rank $W(\Delta)$ is n, replacing the field of complex numbers by the field of rational forms in Δ in Luenberger's procedure [1] solves equation (44) for $T(\Delta)$. Here $F(\Delta)$ is fixed as an $(n-m)\times(n-m)$ diagonal matrix having only negative real entries, none of which is an eigenvalue of $A(\Delta)$. n is the dimension of $x(\Delta)$, and m is the dimension of $y(t)$. As the elements of the diagonal matrix $F(\Delta)$ are negative real numbers, equation (43) is asymptotically stable.

$T(\Delta)$ may have entries which are not polynomials in Δ. Let $\rho(\Delta)$ be the least common multiple of the non-unity denominators of the elements of $T(\Delta)$. Multiply equation (44) by $\rho(\Delta)$ to get

$$\rho(\Delta)\ T(\Delta)\ A(\Delta) - F(\Delta)\rho(\Delta)T(\Delta) = \rho(\Delta)\ G(\Delta)\ C(\Delta). \tag{49}$$

Replacing $T(\Delta)$ by $\rho(\Delta)\ T(\Delta)$ and $G(\Delta)$ by $\rho(\Delta)\ G(\Delta)$, a solution to equation (44) results where all entries are polynomials in Δ. $H(\Delta)$ is now determined from equation (45).

$T(\Delta)$ was determined so that its rows adjoined to the rows of C span n dimensions. This property is not affected by multiplying by $\rho(\Delta)$, therefore equation (46) can be solved for $M(\Delta)$ and $N(\Delta)$. However, $M(\Delta)$ and $N(\Delta)$ may have non-unity denominators. Let $q(\Delta)$ be the least common multiple of the non-unity denominators of $M(\Delta)$ and $N(\Delta)$. Multiplying equation (46) by $q(\Delta)$,

$$q(\Delta)M(\Delta)\ C(\Delta) + q(\Delta)N(\Delta)\ T(\Delta) = q(\Delta)\ I. \tag{50}$$

Replacing $M(\Delta)$ by $q(\Delta)M(\Delta)$, and $N(\Delta)$ by $q(\Delta)N(\Delta)$, equation (47) results, and the pseudo observer relations are satisfied. Q.E.D.

If $q(\Delta)$ does not have a zero root, and all its roots are within the unit circle, an observer can be formed from the pseudo observer by passing the pseudo observer output through the difference equation

$$\hat{x}(t) = [I - \frac{q(\Delta)}{q_o} I]\ x(t) + \frac{M(\Delta)}{q_o}\ y(t) + \frac{N(\Delta)}{q_o}\ z(t). \tag{51}$$

If a zero root occurs, $\hat{x}(t)$ can be reconstructed after a delay, while if roots occur outside or on the unit disk, equation (51) is not stable, and the pseudo observer cannot be converted to an observer.

Sufficient conditions for $q(\Delta)$ to be unity, and theorem 3 to yield an observer directly, are that

$$[C(\Delta)\quad T(\Delta)]'$$

be lower triangular, with only real numbers on the diagonal, or that $C(\Delta)$ have the form

$$C = [I\ 0] \tag{52}$$

and all delay terms in $T(\Delta)$ appear in the first m columns. These results are proven by inverting by elementary operations

$$[C(\Delta)\quad T(\Delta)]'$$

A pseudo observer is used in a closed loop control system by designing a linear state feedback control for the modified system

$$\dot{x}(t) = A(\Delta)\ x(\Delta) + B(\Delta)\ q(\Delta)\ u(t) \qquad (53$$

$$y(t) = C(\Delta)\ x(\Delta) \tag{54}$$

This linear state feedback control has the form

$$u(t) = \int_{-r}^{o} [dk(\theta)]x(t+\theta) \tag{55}$$

where $k(\theta)$ is a matrix of functions of bounded variation. As $q(\theta)$ is bounded and operates equally on each component,

$$q(\Delta)\ u(t) = q(\Delta) \int_{-r}^{0} [dk(\theta)]\ x(t+\theta) = \int_{-r}^{0} [dk(\theta)]\ q(\Delta)\ x(t+\theta) \qquad (56)$$

Hence the feedback control formed from this control and an observer for equations (53) and (54) is equivalent to this feedback control and a pseudo observer for equations (38) and (39), and the responses are identical. Lemma 2 applies, if equations (38) and (39) are replaced by equations (53) and (54).

This section ends with a pseudo observer example. The following time delay system arose in considering yaw control of a slowly taxiing air cushion landing system aircraft:

$$x(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -.005013\Delta^2 & -.02507\Delta^2 & -.5 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ .0005013\Delta \end{bmatrix} u(t) \qquad (57)$$

$$y(t) = [1 \quad 0 \quad 0]\ x(t). \qquad (58)$$

To implement a state feedback control designed using a quadratic criterion, the observer design procedure outlined in theorem 3's proof was used to derive the observer

$$z(t) = \begin{bmatrix} -2 & 0 \\ 0 & -4 \end{bmatrix} z(t) + \begin{bmatrix} 6 + .04513\Delta^2 \\ -56 - .09527\Delta^2 \end{bmatrix} y(t) + \begin{bmatrix} +.0005013\Delta \\ -.0005013\Delta \end{bmatrix} u(t) \qquad (59)$$

where

$$\hat{x}(t) = \begin{bmatrix} 1 \\ 5.5 \\ 5.25-.02507\Delta^2 \end{bmatrix} y(t) + \begin{bmatrix} 0 & 0 \\ .5 & .5 \\ 1.75 & .75 \end{bmatrix} z(t) \qquad (60)$$

This observer gave good results provided a 3 second initial period was allowed for the observer error decay before observer outputs were used to compute the control.

IV. CONCLUSIONS

In this paper, the definition of an observer was generalized from finite dimensional linear systems to linear time delay systems. The qualitative properties of observers are preserved in this generalization. Sufficient conditions are given for constructing identity and reduced order observers, and these conditions are suitable for calculation. Pseudo observers are defined and shown to have properties useful in feedback control synthesis. A sufficient condition for a pseudo observer is given, and the proof of sufficiency provides an algebraic method of designing pseudo observers.

BIBLIOGRAPHY

[1] Luenberger, "Observing the State of a Linear System", IEEE Trans. on Mil. Elec., Vol MIL-8, April 1974, pp 74-80.

[2] Lee, "Identification of Model Parameters from Input Output Data of Dynamic Systems with Time Delays", PhD Dissertation, Minnesota Un., Minneapolis, Minn., August 1970.

[3] Hewer and Nazaroff, "Observer Theory for Delayed Differential Equations", Int. J. Control, Vol 18, No 1, 1973, pp 1-7.

[4] Hale, "Functional Differential Equations", Springer-Verlag, New York, 1971.

[5] Bellman and Cooke, "Differential-Difference Equations", Academic Press, New York, 1963.

[6] Ross and Flugge-Lotz, "An Optimal Control Problem for Systems with Differential-Difference Equation Dynamics", Siam J. Control, Vol 7, No 4, November 1969, pp 609-623.

[7] Luenberger, "Observers for Multivariable Systems", IEEE Trans. Auto. Control, Vol AC-11, No 2, April 1966, pp 190-197.

[8] Gressang, "Observers for Infinite Dimensional Linear Systems", PhD Dissertation, Air Force Institute of Technology, W.P. AFB, Ohio, June 1974.

[9] Hahn, "Theory and Application of Liapunov's Direct Method", Prentice-Hall, Englewood Cliffs, N.J., 1963.

[10] Brockett, "Finite Dimensional Linear Systems", Wiley, New York, 1970.

[11] Lasalle and Lefschetz, "Stability by Liapunov's Direct Method", Academic Press, New York, 1961.

[12] Porter, "Modern Foundations of Systems Engineering", MacMillan, New York, 1966.

[13] Tzafestas and Paraskevopoulos, "On the Decoupling of Multivariable Control Systems with Time Delays", Int. J. Control, Vol 17, No 2, 1973, pp 405-415.

[14] Kalman, Falb, and Arbib, "Topics in Mathematical System Theory", McGraw-Hill, New York, 1969.

ON THE ORDER DISCRIMINATION OF LINEAR DISCRETE-TIME SYSTEM MODELS USING THE INSTRUMENTAL VARIABLE APPROACH

NARESH N GUPTA*
Department of Systems Analysis
Miami University
Oxford, Ohio 45056

ABSTRACT

This paper presents a new method for the order discrimination of linear discrete-time system models with noisy observations. The order discrimination problem is viewed as a sequence of hypothesis tests; at the nth step, a choice is made between models of order n and (n+1). The sequence is terminated the first time that the lower order hypothesis is preferable. The decision criterion is the rank of a matrix which is the product of the system observation matrix and an instrumental variable matrix.

1. INTRODUCTION

The order discrimination problem falls under the general area of system identification. In many applications the system order is not know a priori and must be determined along with the model. Uses for determination of system order by itself might include medical diagnosis, where for instance a model for a healthy lung might be of a different order than that for a diseased one.

Specifically, the problem solved here is the order discrimination of a linear discrete time system model with noisy observations. The order discrimination problem is viewed as a sequence of hypothesis tests, at the nth step, a choice is made between models of order n and (n+1). The sequence is terminated the first time that the lower-order hypothesis is preferable.

This approach has been used by many authors [1,2,3,4] in time series analysis where the discrimination problem reduces to choosing autoregressive (A.R.) and moving average (M.A) processes and deciding on the appropriate order. At the nth step, the decision criteria are the (n+1)th order partial and multiple correlation coefficients respectively.

Eykhoff [5,6] and Sage [7] present fairly detailed surveys of the principal methods of system identification. These procedures differ mainly in the criteria used for defining optimality and the use of a priori information. The starting point of all these techniques is the assumption of the order or the number of parameters of the system to be identified and are, therefore, rightly termed as parameter estimation techniques. It is important to point out that order discrimination and parameter estimation are complementary aspects of the system identification problem. We are at variance with the view that one first estimates the parameters, and then does diagnostic checking to test the goodness of fit of the approximation to the system order. It is difficult to obtain good estimates of the system parameters until the order has been ascertained, and this is precisely the problem tackled in this paper.

---

*Formerly with department of Systems Engineering, Wright State University, Dayton, Ohio 45431

The same could be said of the stochastic approximation technique [8]. It is very doubtful that the stochastic approximation technique will, in general, converge in the case when the assumed order is not the actual order of the system. It should be pointed out that the stochastic approximation technique is based on the 'steady-state' operation of the unknown system and ignores the effects of starting transients. Hence, if it is applied, starting at the time the plant starts up, the procedure is not effective. Furthermore the complexity of the parameter estimation equations will be greatly reduced, if the order of the system is predetermined.

So far, only Gupta [9,10] and Woodside [11] have directly tackled the order determination problem. Woodside [11] developed a procedure on a largely intuitive basis, using an enhancement of the measured product-moment matrix and gives intuitive rather than rigorous tests. Gupta [9,10] developed a procedure for the order discrimination of a linear differential systems with noisy output data. The problem solved by Gupta [9,10] and Woodside [11] is the order determination of a linear differential system characterized by a rational transfer-function with numerator equal to unity. Here we develop a procedure for solution of the order discrimination of a linear differential system characterized by a general rational transfer-function and with noisy observations.

The use of instrumental variables was first suggested by Reiersol [11,12] for the case in which economic variables subject to exact relationships are affected by random disturbances or measurement erros. In this paper, the method is applied to the order discrimination of linear discrete-time system models. The approach is direct in that the model parameters are not estimated at each step. Rather, the test is based on a measure of the rank of a matrix which is the product of an instrumental variable matrix and the observation matrix. The rank is determined using a novel iterative scheme [14,15]. This rank test is more accurate than the conventional methods. A characteristic feature of the instrumental variable method is that it yields consistent estimates even when the observation matrix is correlated with the noise vector.

2. CLASS OF SYSTEMS TO BE CONSIDERED

In this paper, we will be limited to the investigation of a class of linear systems that can be modelled, using a constant coefficient difference equation of the form:

$$\sum_{k=0}^{n} a_{ik} y_i(t-k) = \sum_{k=0}^{m} b_{ik} x(t-k), \; n \geq m \text{ and the} \qquad (1)$$

values of n and m are minimal, consistent with input-output information. The above system model will be denoted as SM_i where the subscript i refers to the ith system model.

In the above, a_{ik}'s and b_{ik}'s are constants; $y(t)$ is the output of the system and $x(t)$ is the input. Without loss of generality, a_n can be assumed to be unity and any or all of the b_{ik}'s may be zero in a particular case.

3. NOTATION

Define $M_N(n,m,s) \equiv$

$$\begin{bmatrix} s(n-1 & \ldots & s(0) & x(n) & \ldots & x(n-m) \\ s(n) & \ldots & s(1) & x(n+1) & \ldots & x(n-m+1) \\ \vdots & & \vdots & & & \vdots \\ s(n+N-2) & \ldots & s(N-1) & x(n+N-1) & \ldots & x(n-m+N-1) \end{bmatrix}_{N\times(n+m+1)} \tag{2}$$

Define $M_N^1(n,m,s) \equiv$

$$\begin{bmatrix} s(n) & s(n-1) & \ldots s(0) & x(n) & \ldots & x(n-m) \\ s(n+1) & s(n) & \ldots s(1) & x(n+1) & \ldots & x(n-m+1) \\ \vdots & & \vdots & \vdots & & \vdots \\ s(n+N-1) & s(n+N-2) & s(N-1) & x(n+N-1) & \ldots & x(n-m+N-1) \end{bmatrix}_{N\times(n+m+2)} \tag{3}$$

$$\theta_i \equiv [-a_{i1} \ldots -a_{in} b_{io} \ldots b_{in}]^T \tag{4}$$

$$G(t,n,m) \equiv [y_i(t-1), y_i(t-2) \ldots y_i(t-n) x(t) \ldots x(t-m)] \tag{5}$$

Now equation (1) can be rewritten as:

$$y_i(t) = G(t,n,m)\theta_i \tag{6}$$

For the system model SM_o, we have

$$z(t) = y_o(t) + w(t) \tag{7}$$

Define $w_n(n) \equiv$

$$\begin{vmatrix} w(n) \;\ldots\; w(n-1) \;\ldots w(0) \\ w(n+1)\ldots w(n-2)\ldots 2(1) \\ \vdots \qquad \vdots \qquad \vdots \\ w(n+N-1) \;\; w(n+N-2)\ldots w(N-1) \end{vmatrix}_{N\mathrm{x}(n+1)} \tag{8}$$

4. DEFINITIONS

Consider the system model SM_o represented by equation (1) with the subscript i=0, and only the noisy output z(t) given by equation (1) is available for observation. Given an input sequent $\{x(t)\}_{t=0}^{\infty}$, a sequence of Nx(n+m+1) matrices $\{V_N\}_{N\geq n+m+1}^{\infty}$ is an instrumental matrix sequence if the following probability limits hold:

$$\underset{N\to\infty}{\mathrm{Plim}} \; \frac{1}{N} V_N^T W_N(n) = 0 \tag{9}$$

$$\underset{N\to\infty}{\mathrm{Plim}} \; \frac{1}{N} V_N^T M_N(n,m,y_o) \text{ exists and is non-singular} \tag{10}$$

The elements of the matrices V_N, $N\geq n+m+1$, are called instrumental variables.

5. STATEMENT OF THE PROBLEM

Consider the nth order system model SM_0 represented by equation (1). Suppose that only the input x(t) and the noisy output z(t) are available for measurement. The aim of this paper is to show that given SM_0 is stable and has no pole zero cancellation, then for a particular class of inputs and certain weak conditions on w(t), the order of the system can be determined using Theorems 1, 2 and 3. Further any stable nth order system model SM_i, represented by equation (1), which has no pole zero cancellation can be used to generate a matrix sequence which is an instrumental variable matrix sequence [16].

For ease of reference, the following conditions are listed: Condition 1) SM_i represented by equation (1) is a stable, nth order, discrete time linear model. Condition 2) Let $X(t) \equiv [x(t+n), x(t+n-1), \ldots x(t)]_{(n+1)\mathrm{x}1}$ where $\{x(t)\}_{t=0}^{\infty}$ satisfies the condition

$$\underset{N\to\infty}{\mathrm{Lt}} \; \frac{1}{N} \sum_{t=0}^{N-1} X(t)X^T(t) = \lambda I \text{ where } \lambda \text{ is non-zero constant}$$

Condition 3) $\{x(t)\}_{t=0}^{\infty}$ is the input for both SM_0 and SM_i.

Condition 4) $|x(t)|$ is uniformly bounded.

Condition 5) $z(t)=y_0(t)+w(t)$ where $\{w(t)\}_{t=0}^{\infty}$ is a sample from a zero mean stationary process whose convariance function $R_w(t)$ satisfies $\lim\limits_{t\to\infty} tR_w(t)=0$.

Condition 6) θ_i is statistically independent of the noise process $w(t)$.

Condition 7) Both SM_0 and SM_i have no pole zero cancellation.

6. SOLUTION OF THE PROBLEM

Since the proofs are lengthy, we state here the main results in the form of Theorems without proofs:

Theorem 1:

Given the conditions 1 to 4, and 7,

(a) the matrix $\lim\limits_{N\to\infty} \frac{1}{N}M_N^T(n,m,y_i)M_N(n,m,y_o)$

exists and is nonsingular.

(b) The matrix $\lim\limits_{N\to\infty} \frac{1}{N}[M_N^1(n-1,m,y_i)]^T M_N^1(n-1,m,y_o)$

exists and is nonsingular.

(c) The matrix $\lim\limits_{N\to\infty} \frac{1}{N}[M_N^1(n,m,y_i)]^T M_N^1(n,m,y_o)$

exists and is singular.

Theorem 2:

Given the conditions 1 to 7, the matrix $\frac{1}{N}M_N^T(n,m,y_i)$ $M_N(n,m,z)$ is consistent and (hence) asympotically unbia biased estimate of the matrix $\frac{1}{N}M_N^T(n,m,y_i)M_N(n,m,y_0)$.

Order Test.

The above Theorems can be implemented to give the following sequential test:

Check if the matrix $\frac{1}{N}[M_N^1(n,m,y_i)]^T M_N^1(n,m,z) \equiv F(n,m)$ is singular. If it is singular, then the orders of the system model are n and m. If it is non-singular, then increase n or m or both and recheck if the new matrix is non-singular and so on.

The following Theorem [14,15] is used to test the rank of the matrix $F(n,m)$.

Theorem 3:

The trace of $[I-\alpha FF^T]^{2k}$ is monotonically decreasing function of k, $k=1,2,\ldots$, converging to the nullity of matrix F with $0<\alpha<2/\lambda_1$ where λ_1 is the maximum eigenvalue of FF^T (or F^TF).

Remark: In using the above Theorem, it is not necessary to compute λ_1, the maximum eigenvalue of FF^T (or F^TF). Writing $FF^T=(g_{ij})$, $i,j=1,2,\ldots,(n+m+2)$, we conclude from the Gershgorin Theorem that

$$\lambda_1 \leq \max \left[\sum_{j=1}^{n+m+2} |g_{ij}| \right]$$

and the condition on α can be replaced by

$$0<\alpha< \left[\frac{2}{\max \sum_{j=1}^{n+m+2} |g_{ij}|} \right]$$

7. CONCLUDING REMARKS

The method presented is especially useful in determining the order of very high order systems. Presently, a series of detailed digital computer studies are being carried out to determine the effectiveness of the proposed order discrimination scheme. It is anticipated that the present method will give reasonably accurate model order under considerably large noise levels (perhaps up to a level of .25 noise to signal ratio). Once the order of the system has been determined, the parameters of the system can also be determined using the instrumental variable method [16,17,18]. It should be noted that all the results listed so far depend on the use of asymptotic approximations. The order of the magnitude of the errors involved in this approximation is found [12] to be proportional to the number of instrumental variables, so that, if the asymptotic approximations are to be used, this number must be small. The above points out the need for determining the sustem order before implementing even the instrumental variable method for system parameter estimation.

8. ACKNOWLEDGEMENT

The author would like to thank Miami University for the support of this work under a 1974 Faculty Summer Research Fellowship.

REFERENCES

[1] A. Rudra, Discrimination in Time Series Analysis, Biometrika 39 (1952) pp. 434-439.

[2] P. Whittle, Tests of Fit in Time Series, Biometrika 39 (1952) pp. 290-318.

[3] M. H. Quenouille, A Large Sample Test for the Goodness of Fit of Autoregressive Schemes, J. Royl. Stat. Soc. 116 (1947) pp. 123-129.

[4] G. M. Jenkins, Tests of Hypothesis in the Linear Autoregressive Model II, Biometrika 43 (1956) pp. 186-199.

[5] Peter Eykhoff, Process Parameter and State Estimation, Identification in Automatic Control Systems IFAC Symp. (1967) Sect. 2, pp. 1-15.

[6] Peter Eykoff and K. J. Astrom, System Identification-A Survey, Automatica 7 (1971) pp. 123-162.

[7] A. P. Sage and M. Cuenod, Comparison of Some Methods Used for Process Identification, Identification in Automatic Control Systems IFAC Symp. (1967) Sec. 1, pp. 1-15.

[8] D. J. Sakrison, The Use of Stochastic Approximation to Solve System Identification Problem, IEEE Trans. on AC, 12 (1967) pp. 86-87.

[9] Naresh N. Gupta, The Order Determination of Linear Differential System With Noisy Observations, Proc. of the Int. Conf. on Microelectronics, Circuits and System Theory, Sidney, Australia (Aug, 1970) pp. 77-87.

[10] Naresh N. Gupta, On the Order Discrimination of Linear Systems, Proc. 1971, IEEE Symposium on Decision and Control; Int. J. Systems Science, Vol. 4, No. 3, pp 473-484 (1973)

[11] C. M. Woodside, Estimation of the Order of Linear Systems, Automatica 7 (1971), pp. 727-733.

[12] J. D. Sargan, The Estimation of Economic Relationships Using Instrumental Variables, Econometrica, Vol. 26 (1958) pp. 393-415.

[13] O. Reiersol, Confluent Analysis by Means of Lag Movements and Other Methods of Confluence Analysis, Econometrica, Vol. 9 (1941) pp. 7-23.

[14] Naresh N. Gupta, On the Convergence of an Iterative Method for the Computation of Generalized Inverse and Associated Projections, Int. J. of Systems Science, 1971 Vol. 2, No. 1, pp. 67-75.

[15] Naresh N. Gupta, An Optimum Iterative Method for Computation of Matrix Rank, Proc. Ninth Annual Allerton Conf. Circuits and Systems Theory, Urbana, Illinois (Oct., 1971) and IEEE Trans. on System, Man and Cybernetics, Vol. 2 (1972) pp. 437-438.

[16] B. Finigan and I. H. Rowe, On the Identification of Linear Discrete Time System Models Using the Instrumental Variable Method, Proceedings the 3rd IFAC Symposium on Identification and System Parametric Estimation (1973) paper classification TM-5, pp. 729-735.

[17] K. Wang and E. Polek, Identification of Linear Discrete Time System Using the Instrumental Variable Method, IEEE Trans. A.C., Vol. 12(1967) pp. 707-718.

[18] P. C. Young, An Instrumental Variable Method for Real Time Identification of a Noisy Process, Automaticia, Vol. 6 (1970) pp. 271-287.

DECOUPLING OF LINEAR SYSTEMS BY OUTPUT FEEDBACK

N. VISWANADHAM and D. P. ATHERTON
Dept. of Electrical Engineering, University of New Brunswick
Fredericton, New Brunswick, Canada

ABSTRACT

Necessary and sufficient conditions for decoupling a class of linear multivariable systems using output feedback are derived. When the given system does not satisfy these conditions, a compensator is presented which allows decoupling of the augmented system using output feedback. Finally the question of open loop decoupling using dynamic compensators is considered. Examples are presented to illustrate the theory and to discuss the stability of the decoupled systems.

1. INTRODUCTION

Let $\mathcal{S} = (A,B,C)$ be a linear time-invariant system described by the equations

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t)\end{aligned} \tag{1}$$

where x(t), u(t) and y(t) are n,m and m dimensional column vectors and A,B,C are n x n, n x m and m x n matrices respectively. A considerable amount of literature exists on the problem of decoupling (1) by state variable feedback of the type $u = Fx + Gw$ where F and G are constant matrices and w is a reference input vector. A complete solution to this problem has been given by Falb and Wolovich [1] and Gilbert [2]. For the current status on state feedback decoupling and stabilization of the decoupled systems, the reader is referred to the recent article by Wonham and Morse [3].

In practice all the state variables of a system are not usually available for measurement. Hence it is of significant interest to develop procedures for decoupling using only output feedback. Silverman and Payne [4] accomplish this by using an observer to estimate x from y and u. Howze and Pearson [5] add additional dynamics to the system (1) and replace the state feedback by output feedback. These approaches involve a two step procedure of first finding the state feedback gain matrix to achieve the required objective and then replacing it by output feedback. The problem considered in this paper is to find conditions under which decoupling is possible directly by output feedback for a certain class of systems.

Falb and Wolovich [1] and Gilbert [2] considered this problem as a special case of the state variable feedback problem, however, their procedure is such that no solution may be possible and the conditions which must be satisfied to obtain a solution are not given. Recently the problem has also been considered by Howze [6] and Denham [7] who derive the necessary and sufficient conditions for decoupling using output feedback. For the class of systems considered in this paper the conditions derived here are more clear and easier to check than those given in [6] and [7]. Further, the approach presented leads to other significant results on the design of compensators to allow output feedback decoupling and on the open loop decoupling problem.

In Section 3, a necessary and sufficient condition for obtaining output feedback decoupling is presented. When this condition is not satisfied it is shown in section 4, that a dynamic compensator can be designed such that the augmented system is decouplable using output feedback.

By appropriately modifying the results of section 4, one can find a realizable compensator so that system (1) is decoupled in an open loop fashion. This is dealt with in section 5. A practical example is presented at the end of section 4, to illustrate the theory.

2. NOTATION AND PRELIMINARY RESULTS

For the system described by (1), define the indices d_i, 1 = 1,2,..,m, the vectors D_i and the matrix T by

$$d_i = \min\{j: C_i A^j B \neq 0, j = 0,1,2..(n-1)\} \tag{2}$$

$$D_i = C_i A^{d_i} B \tag{3}$$

and

$$T' = [C'_1, A'C'_1, .., (A')^{d_1} C'_1, C_2, ..., (A')^{d_m} C'_m] \tag{4}$$

where D_i and C_i are the i th rows of the matrices D and C respectively. It is well known that [1,2] system (1) is decouplable by state variable feedback of the type

$$u = Fx + Gw \tag{5}$$

where G is nonsingular and w is a m x 1 reference input vector, if and only if $\det D \neq 0$. The class of systems considered in this paper are those which satisfy the conditions $\sum_{i=1}^{m} d_i + m \triangleq d = n$ and $D \neq 0$. Even though these conditions appear restrictive, there are a significant number of physical systems that fall into this class[†].

Before proceeding to the main result several auxiliary lemmas are given.

Lemma 1: Let d = n and $\det D \neq 0$. Then system (1) is completely controllable and observable. Furthermore the observability is invariant under state variable feedback of the type (5).

Proof: It is well known [1,2] that the two sets of d vectors $\{C_1, ..., C_1 A^{d_1}, C_2 ..., C_m A^{d_m}\}$ and $\{b_1, ..., A^{d_m} b_m\}$, where b_i is the i th column of B, are linearly independent when D is nonsingular. Also it is true that (see[1]) $C_i (A + BF)^j = C_i A^j$; $j \leqslant d_i$, i = 1,2,..m. Since d = n, the lemma follows.

Lemma 2: Let d = n and $\det D \neq 0$. Then any F that decouples (1) must be of the form $F = -D^{-1}H$ where H_i, the ith row of H is given by

$$H_i = C_i A^{d_i+1} + \sum_{j=0}^{d_i} \alpha_{ij} C_i A^j \tag{6}$$

where the α_{ij}'s are arbitrary constants.

Proof: See Mufti [8,9].

Lemma 3: Let d = n. Then the system (1) is invertible if and only if $\det D \neq 0$.

Proof: Let $\det D = 0$, then to prove that the system is not invertible, first obtain the equations $\dot{x} = Ax + Bu$, $y^* = \Delta_1 x + Du$ by repeated differentiation of the equation $y_i = C_i x$ and using (2) where $\Delta'_1 = [(C_1 A^{d_1+1})' ... (C_m A^{d_m+1})']$ and $y^{*\prime} = [y_1^{(d_1+1)}, ... y_m^{(d_m+1)}]$. Now performing Silverman's algorithm [11] on these equations, it can be noted that rank of the coefficient matrix of u will not improve as the iterations proceed. The lemma is proved by noting that (1) is invertible when $\det D \neq 0$ [10].

[†] See for example the practical example in Section 7 of [1], Montgomery's aircraft model for lateral control design [12], the three degree of freedom linearized model of an aircraft [13], and the linearized model of an aircraft gas turbine [14].

3. DECOUPLING USING OUTPUT FEEDBACK

In this section, the problem of decoupling the class of systems defined in section 2 using feedback of the type

$$u = Ky + Gw \tag{7}$$

is considered. As observed by Falb and Wolovich [1], this problem has a solution if and only if F can be expressed as F = KC for some K where F is the decoupling state feedback matrix.

Theorem 1: Let $d = n$ and $\det D \neq 0$. Then (1) can be decoupled using output feedback if and only if for each i, $i = 1,2,\ldots,m$, $C_i A^{(d_i+1)}$ can be expressed as a linear combination of the $(d_i + m)$ row vectors $C_i A^{d_i}, \ldots C_1 A, C_i, \ldots C_m$.

Proof: Suppose $C_i A^{d_i+1}$, $i = 1,2\ldots m$, can be expressed as a linear combination of the $(d_i + m)$ vectors indicated above, then there exists constants α_{ij}'s and β_{ij}'s such that

$$C_i A^{d_i+1} = \sum_{j=0}^{d_i} \alpha_{ij} C_i A^j + \sum_{\substack{k=1 \\ k\neq i}}^{m} \beta_{ik} C_{ik} = \sum_{k=1}^{d_i} \alpha_{ij} C_i A^j + \sum_{k=1}^{m} \beta_{ik} C_k + (\alpha_{i0} - \beta_{ii}) C_i \tag{8}$$

or

$$\bar{H}_i \triangleq C_i A^{d_i+1} - \sum_{j=1}^{d_i} \alpha_{ij} C_i A^j - (\alpha_{i0} - \beta_{ii}) C_i = \sum_{k=1}^{m} \beta_{ik} C_k \tag{9}$$

where the β_{ii}'s are arbitrary and $\bar{H}_i$ is the i th row of the matrix $\bar{H}$.

Now let

$$\beta_i = [\beta_{i1}, \ldots. \beta_{im}] \text{ and } \beta' = [\beta_1', \beta_2', \ldots.. \beta_m'] \tag{10}$$

From lemma 2 it follows that $u = D^{-1}\beta y + D^{-1} w$ is the output feedback decoupling law. Suppose the system (1) is decoupled using output feedback (7). Again by lemma 2, $KC = -D^{-1}H$ where H is defined by (6). Substituting for H and rearranging it can be seen that $C_i A^{d_i+1}$ is expressible as a linear combination of the $(d_i + m)$ row vectors indicated in the theorem Q.E.D.

Observe from the proof of theorem 1 that the diagonal elements of β are arbitrary and the rest are fixed. A similar observation was made Howze [6]. Arbitrary pole placement in the decoupled system is in general not possible though the system may be stabilizable. The decoupled system equations are given by $y_i^{(d_i+1)} + \sum_{j=1}^{d_i} \alpha_{ij} y^{(j)} - (\alpha_{i0} - \beta_{ii}) = w_i$. Clearly the system is stabilizable if for some β_{ii}, all the zeros of the polynomial $\lambda^{d_i+1} - \sum_{j=1}^{d_i} \alpha_{ij} \lambda^j - (\alpha_{i0} - \beta_{ii})$ lie in the left half of the complex plane.

Example 1: Consider the system

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -2 & a_1 & 4 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & a_2 \end{bmatrix} ; \quad B = \begin{bmatrix} 0 & 0 \\ 1 & 1 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} ; \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

By direct calculation, it is easy to see that $d_1 = d_2 = 1$; $d = 4 = n$ and

$$D = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} ; \quad \begin{aligned} C_1 A^2 &= a_1 C_1 A - 2C_1 + 4C_2 \\ C_2 A^2 &= a_2 C_2 A + C_1 \end{aligned}$$

Hence a choice of $G = D^{-1}$ and $K = D^{-1}\beta$ decouples the system. The matrix K and the decoupled system matrix are given by

$$K = -\bar{D}^{-1}\begin{bmatrix}\beta_{11} & 4 \\ 1 & \beta_{22}\end{bmatrix} ; A + BKC = \text{diag}\left(\begin{bmatrix}0 & 1 \\ -2-\beta_{11} & a_1\end{bmatrix}, \begin{bmatrix}0 & 1 \\ -\beta_{22} & a_2\end{bmatrix}\right)$$

It can be seen that the system is stabilizable by a proper choice of β_{11} and β_{22} if a_1 and a_2 are both negative. If anyone of them is positive however, the decoupled system will be unstable.

4. DECOUPLING USING DYNAMIC OUTPUT FEEDBACK

In this section a procedure for designing a dynamic precompensator for systems which are not decouplable by constant output feedback (7) is presented. Suppose the system (1) satisfies the conditions d = n and $\det D \neq 0$ but the vectors $C_i A^{d_i+1}$, $i = 1,2,\ldots m$ cannot be expressed as a linear combination of the (d_i+m) vectors $C_i A^{d_i},\ldots C_i A, C_1,\ldots C_m$. Then decoupling by constant output feedback is not possible. In this instance, a precompensator of appropriate dimension can be designed so that the augmented system meets all the specifications of theorem 1 and thus can be decoupled by output feedback.

Suppose the plant (1) is augmented by a compensator of order k

$$\dot{z} = J_1 z + L_1 v \quad ; \quad u = M_1 z + N_1 v \tag{11}$$

where v is the (m x 1) compensator input vector, the order k and the matrices J_1, L_1, M_1 and N_1 are to be determined.

The aim is to generate an augmented system defined by

$$\dot{\tilde{x}} = \begin{bmatrix}\dot{x} \\ \dot{z}\end{bmatrix} = \begin{bmatrix}A & BM_1 \\ 0 & J_1\end{bmatrix}\begin{bmatrix}x \\ z\end{bmatrix} + \begin{bmatrix}BN_1 \\ L_1\end{bmatrix} v = \tilde{A}\tilde{x} + \tilde{B}v \ ; \ y = \tilde{C}\tilde{x} = [C \ 0]\tilde{x} \tag{12}$$

that satisfies the conditions (i) $\sum_{i=1}^{m}\tilde{d}_i + m = n + k$ (ii) $\det \tilde{D} \neq 0$ and (iii)

$$\tilde{C}_1 \tilde{A}^{\tilde{d}_i+1} = \sum_{j=1}^{d_i}\tilde{\alpha}_{ij}\tilde{C}_i\tilde{A}^j + \sum_{j=1}^{m}\tilde{\beta}_{ij}\tilde{C}_j \tag{13}$$

where $\tilde{d}_i$ and $\tilde{D}$ are defined for (12) in a similar manner to (2) and (3).

To proceed, define indices p_i, $i = 1,2,\ldots m$, the first integers such that

$$C_i A^{p_i} = \sum_{j=1}^{p_i-1}\alpha_{ij}C_i A^j + \sum_{j=1}^{m}\beta_{ij}C_j \tag{14}$$

Note that $p_i \geqslant d_i + 1$ and $p \triangleq \sum_{i=1}^{m} p_i > n$. Now it will be shown that it is possible to augment the system (1) by means of a compensator of order $k = p - n$ in such a way that the augmented system is decouplable by output feedback. Define, the (p x n) and (p x p) matrices R and S by

$$R' = [R_1', R_2', \ldots, R_m'] \ ; \ S' = [S_1', S_2', \ldots, S_m'] \tag{15}$$

where

$$R_i' = [C_i', (C_i A)', \ldots, (C_i A^{p_i-1})'] ; \ S_i = [R_i, E_i] \tag{16}$$

for some p_i x (p-n) matrix E_i. The p x (p-n) matrix E defined by $E = [E_1', E_2', \ldots, E_m']$ is chosen such that S is nonsingular and is constructed according to the following lemma.

Lemma 5: Given R, there always exists a matrix E such that S is nonsingular.

Proof: First note that by lemma 1, $C_i A^j$, $i = 1,2,\ldots m$, $j = 0,1,\ldots d_i$ are linearly independent rows of R. Rearrange the rows of R such that the first d rows are $C_1, C_1 A, \ldots C\, A^{d_1}, C_2, \ldots C_2 A^{d_2}, \ldots C_m A^{d_m}$. Thus R is now equivalent to R where $\bar{R} = [\bar{T}', \bar{R}_2']$ with rank T = d = n where T is as defined in (4). Now let

$$\bar{S} = \begin{bmatrix} \bar{T} & 0 \\ R_2 & I \end{bmatrix} \tag{17}$$

Thus rank $\bar{S} = p$ and by rearranging the rows of $\bar{S}$, S and E are obtained.

Q.E.D.

From lemma 5, observe that E_i has the following structure

$$E_i = \begin{bmatrix} & 0 & \\ \underline{0} & I_{p_i-d_i-1} & \underline{0} \end{bmatrix} \tag{18}$$

and that $\tilde{C}_i$ is the first row of S_i, $i = 1,2,\ldots,m$. Now M_1 and J_1 will be determined such that S_i contains $\tilde{C}_i, \tilde{C}_i\tilde{A}, \ldots \tilde{C}_i\tilde{A}^{p_i-1}$ as its rows.

From (16) it is true that there exists a (p x p) matrix Q such that

$$RA = QR \tag{19}$$

where $Q_{ij}, i,j = 1,2,..m$ is a $p_i \times p_j$ matrix given by

$$Q_{ii} = \left[\begin{array}{c|cccc} \underline{0} & & I_{p_i-1} & & \\ \hline \beta_{ii} & \alpha_{i1} & \alpha_{i2} & \cdots & \alpha_{i,p_i-1} \end{array}\right] \quad ; \quad Q_{ij} = \begin{bmatrix} \underline{0} & \\ & \underline{0} \\ \beta_{ij} & \end{bmatrix} \tag{20}$$

Now choose

$$S \begin{bmatrix} BM_1 \\ J_1 \end{bmatrix} = QE \tag{21}$$

Combine equations (19) and (21) and use (12) to yield

$$S\tilde{A} = QS \tag{22}$$

It can be easily seen from (22) that $\tilde{\mathcal{S}}$ indeed satisfies (13) with $\tilde{d}_i = p_i - 1$, $\tilde{\alpha}_{ij} = \tilde{\alpha}_{ij}$ and $\tilde{\beta}_{ij} = \beta_{ij}$. To show that S_i contains $\tilde{C}_i, \ldots \tilde{C}_i\tilde{A}^{p_i-1}$ as its rows first define $\bar{C}$ by $\bar{C} = \text{diag}(\bar{C}_1, \bar{C}_2, \ldots \bar{C}_m)$ where $C_i = [1,0,\ldots 0]$ is $1 \times p_i$ vector and note that $\tilde{C}S^{-1} = \bar{C}$. (This follows since $\tilde{C}_i$ is the first row of S_i, $i = 1,2,\ldots m$.) Now from (22), it can be concluded that $\bar{\mathcal{S}} = (Q,\bar{B},\bar{C})$ is an equivalent representation of $\tilde{\mathcal{S}} = (\tilde{A},\tilde{B},\tilde{C})$ obtained by making a non-singular coordinate transformation $\bar{x} = S\tilde{x}$ on (12) where $\bar{B} = S\tilde{B}$. For the system $\bar{\mathcal{S}}$, by direct calculation, it can be shown that for each $i=1,2,\ldots,m$

$$\begin{bmatrix} \bar{C}_i \\ \bar{C}_i Q \\ \bar{C}_i Q^{p_i-1} \end{bmatrix} = \Big[\underline{0} \quad \underbrace{I_{p_i-1}}_{i \text{ th block}} \quad \underline{0}\Big] \tag{23}$$

From (23) and the observation $\tilde{C}_i\tilde{A}^j = C_iQ^jS$, $i = 1,2,..m$, $j = 0,1\ldots(p_i-1)$, it follows that $\tilde{C}_i\tilde{A}^j$, $j = 0,1,..p_i-1$ are the rows of S_i.

Now choose

$$S\tilde{B} = S\begin{bmatrix} BN_1 \\ L_1 \end{bmatrix} = \text{diag}\,(\bar{b}_1, \bar{b}_2, \ldots \bar{b}_m) \tag{24}$$

where the $(p_i \times 1)$ vector $\bar{b}_i$ is given by $\bar{b}_i' = [0,0,\ldots,1]$. From (24) and the fact that $\tilde{C}_i\tilde{A}^j$, $j = 0,1,\ldots p_i-1$ are the rows of S_i, it can be concluded that $\tilde{d}_i = p_i-1$ and $\tilde{D} = I$.

Thus far, it has been shown that when the compensator parameters satisfy (21) and (24), an augmented system that is decouplable by output feedback can be generated. It remains only to solve for M_1, N_1, J_1 and L_1 from (21) and (24). It can be shown that (21) yields a unique solution for M_1 and J_1. To see this, firstly premultiply (21) with a permutation matrix $\mathcal{P}$ and use (22) to yield

$$\mathcal{P}S\begin{bmatrix}BM_1\\ J_1\end{bmatrix} = \bar{S}\begin{bmatrix}BM_1\\ J_1\end{bmatrix} = \begin{bmatrix}T & 0\\ \bar{R}_2 & I\end{bmatrix}\begin{bmatrix}BM_1\\ J_1\end{bmatrix} = \begin{bmatrix}I & 0\\ \bar{R}_2T^{-1} & I\end{bmatrix}\begin{bmatrix}TBM_1\\ J_1\end{bmatrix} = \mathcal{P}QE \tag{25}$$

From (25) and (26) and the structure of $\mathcal{P}$ one gets $\mathcal{P}Q = \begin{bmatrix}\bar{Q}_1\\ \bar{Q}_2\end{bmatrix}$;

$$Q_1 = \text{diag}(\bar{Q}_{11},\bar{Q}_{22},..,\bar{Q}_{mm});\ \bar{Q}_{ii} = [0\ I_{d_i+1}\ 0] \tag{26}$$

where $\bar{Q}_1$ and $\bar{Q}_{ii}$ are $(n \times p)$ and $(d_i+1) \times p_i$ matrices respectively. Now using (18), one can determine $\bar{Q}_1E$ as

$$\bar{Q}_1E = \text{diag}(E_{11},E_{22},...,E_{mm})\ ;\ E'_{ii} = \left[0\ \ 0\cdots\underset{0}{\cdot}0\ 1\right] \tag{27}$$

where E_{ii}, $i=1,2,...,m$ is a $(d_i+1) \times (p_i-d_i)$ matrix. Also using (2) and (3) TB can be calculated as $TB = \text{diag}(t_1,t_2,...t_m)$. D where the (d_i+1) column vector is given by $t'_i = [0,0,...,1]$. From (21) and (25) it follows that

$$J_1 = (-\bar{R}_2\bar{T}^{-1}\bar{Q}_1 + \bar{Q}_2)\ E\ ;\ TBM_1 = \bar{Q}_1E \tag{28}$$

From (2), (3) and (25) and (28) one gets

$$M_1 = \bar{D}^{-1}\ \text{diag}(m_1,m_2,....m_m)\ ;\ m_i = [1,0,..0] \tag{29}$$

where m_i is a (p_i-d_i-1) row vector. Equation (24) can be solved in a similar way for L_1 and N_1. Also from lemma 1, it follows that the system $\mathcal{G}$ as well as the system decoupled by output feedback, are controllable and observable.

The preceding results provide a constructive proof of the following theorem.

Theorem 2: Let $d = n$ and $\det D \neq 0$. Then it is possible to compensate system(1) by means of a compensator of order $k = p - n$ such that the augmented system is decouplable by output back $v = \tilde{K}y + \tilde{G}w$. Further the decoupled system is controllable and observable.

Observe that each subsystem $i = 1,2..m$ has only p_i-d_i states available for measurement assuming that all the compensator states are measurable. Hence the decoupled system may be stabilizable but arbitrary pole placement may not be possible. Also it can be seen that the characteristic equations of each of the m subsystems is given by

$$s^{p_i} + \sum_{j=1}^{p_i-1} \alpha_{ij}s^j + \beta_{ii} = 0,\ i = 1,2,...,m \text{ where the } \beta_{ii}\text{'s are arbitrary.}$$

A Practical Example: Here, the detailed application of the theory developed in this section to a 4th order linearized model of an aircraft gas turbine engine [14] is presented.

The gas turbine is represented by the linearized equations [14]

$$\frac{d}{dt}\begin{bmatrix}\Delta N_H\\ \Delta N_L\\ \Delta A_J\\ \Delta Q_F\end{bmatrix} = \begin{bmatrix}-1.268 & -0.04528 & 1.498 & 951.5\\ 1.002 & -1.957 & 8.52 & 1240.0\\ 0 & 0 & -10 & 0\\ 0 & 0 & 0 & -100\end{bmatrix}\begin{bmatrix}\Delta N_H\\ \Delta N_L\\ \Delta A_J\\ \Delta Q_F\end{bmatrix} + \begin{bmatrix}0 & 0\\ 0 & 0\\ 10 & 0\\ 0 & 100\end{bmatrix}\begin{bmatrix}\Delta A_d\\ \Delta Q_d\end{bmatrix}$$

The state variables are ΔN_H = High pressure-spool speed, ΔN_L = Low pressure spool speed, ΔA_J = Jet-pipe-nozzle area, ΔQ_F = Fuel-flow rate. The inputs are ΔA_d = demanded jet pipe nozzle area, ΔQ_d = demanded fuel-flow rate. The outputs are the two turbine speeds ΔN_H and ΔN_L. All the variables are expressed as perturbations from an operating point. Observe that

$$D = \begin{bmatrix}C_1AB\\ C_2AB\end{bmatrix} = \begin{bmatrix}14.98 & 95.15\text{x}10^3\\ 85.2 & 124.0\text{x}10^3\end{bmatrix}\ ;\ \bar{D}^{1} = \begin{bmatrix}-1.98424\text{x}10^{-2} & 1.5226\text{x}10^{-2}\\ 1.3634\text{x}10^{-6} & -2.40\text{x}10^{-6}\end{bmatrix}$$

and $\det D = 624.9253\text{x}10^4$. Also $d_1 = d_2 = 1$ hence, $d = 4$. It can be checked that C_1A^2 and C_2A^2 cannot be expressed as a linear combination of C_1A, C_1,C_2 respectively. Thus decoupling by constant output feedback is not possible. However the system can be decoupled using dynamic output feedback. By direct calculation, it is possible to show that $p_1 = p_2 = 3$, $C_1A^3 = \alpha_{11}C_1A + \alpha_{12}C_1A^2 + \delta_{11}C_1 + \delta_{12}C_2$ and $C_2A^3 = \alpha_{21}C_1A + \alpha_{22}C_2A^2 + \delta_{21}C_1 + \delta_{22}C_2$ where $\alpha_{11} = -1167.95$; $\alpha_{12} = -111.5438$; $\delta_{11} = -1308.5070$; $\delta_{12} = -36.9525$; $\alpha_{21} = -1196.513$; $\alpha_{22} = -111.8375$; $\delta_{21} = 845.3933$; $\delta_{22} = -2772.073$.

The order of the compensator is given as $p - n = 6 - 4 = 2$. Now calculate

$$T = \begin{bmatrix} C_1 \\ C_1A \\ C_2 \\ C_2A \end{bmatrix} = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 0.0 \\ -12.68 & -0.4528 & 1.498 & 951.5 \\ 0.0 & 1.0 & 0.0 & 0.0 \\ 1.002 & -1.957 & 8.52 & 1240.0 \end{bmatrix} ; \quad R_2 = \begin{bmatrix} (C_1A^2)' \\ (C_2A^2)' \end{bmatrix} = \begin{bmatrix} 1.5625 & -3.2314 \\ 0.1460 & 3.7845 \\ -17.265 & -100.373 \\ -96412.625 & -125473.25 \end{bmatrix}$$

Also E, $\bar{Q}_1E_2$, $\bar{Q}_2E$ and T^{-1} as given as

$$E = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} : \bar{Q}_1E = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} ; \quad T^{-1} = \begin{bmatrix} 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 \\ -0.404164 & -0.198424 & -0.20812 & 0.15226 \\ 0.0019672 & 0.0013634 & 0.000152 & -0.00024 \end{bmatrix}$$

$$\bar{Q}_2E = \begin{bmatrix} \alpha_{12} & 0 \\ 0 & \alpha_{22} \end{bmatrix}$$

Using (28) and (29), M_1 and J_1 can be obtained as

$$J_1 = \begin{bmatrix} 16.48 & -20.31 \\ 151.154 & -126.4075 \end{bmatrix} ; \quad M_1 = \begin{bmatrix} -1.98424\text{x}10^{-2} & 1.5226\text{x}10^{-2} \\ 1.3634\text{x}10^{-6} & -2.40\text{x}10^{-6} \end{bmatrix}$$

In a similar way one can directly obtain $L_1 = I$ and $N_1 = \underline{0}$.

B. Decoupling using output feedback: To determine the K and G that decouple (12) observe that $D = I$ and $d_1 = 2 = d$. From the results of section 3, it follows that

$$G = I \text{ and } K = \begin{bmatrix} 1308.5070 + \beta_{11} & 36.9575 \\ -845.3933 & 2772.073 + \beta_{22} \end{bmatrix}$$ where β_{ii}, $i = 1,2$ are arbitrary.

C. Stability of the decoupled system: It can be easily seen by using K given above that the sub system characteristic equations are

$s^3 + 111.5438s^2 + 1167.95s + \beta_{11} = 0$ and $s^3 + 111.8375s^2 + 1196.513s + \beta_{22} = 0$.

By applying the Routh-Hurwitz test for the above equations it can be checked that the decoupled system is stable for $0 < \beta_{11} < 111.54 \times 1167.95$ and $0 < \beta_{22} < 111.8375 \times 1196.513$. The root diagrams for these equations can also be drawn easily and β_{11} and β_{22} can be chosen to achieve a desirable transient response.

D. Comparison with other design techniques: As can be seen from the results of the above example, it is possoble to design systems for decoupling and stability using the techniques presented in section 4. The order of the compensator required is two.

It is also possible to decouple and stabilize the plant given here by using state variable feedback [1]. To estimate the state variables, the reduced observer of order two is required. However, if one wishes to use the compensator of Howze and Pearson [6], then the order of the compensator required is 4 = mp, p is the observability index which is two for this system. McMorran [14], however uses a decoupling technique which is close to the technique presented in the following section and presents a fourth order compensator to achieve decoupling.

5. OPEN LOOP DECOUPLING

Here we consider the problem of finding a compensator that decouples (1) in an open loop fashion. Most classical methods of achieving noninteraction involve generating a compensator that decouples the given system and then adding unity feedback to achieve certain specifications (see eg. Peschon [16], Tou [17]and Kavanagh [18]). However, the classical techniques often always lead to complex transfer function matrices for the compensators. Here a procedure us given which yields a realizable compensator that decouples the given system.

Suppose, we augment a compensator of order δ defined by

$$\dot{z} = J_2 z + L_2 v \quad ; \quad u = M_2 z + N_2 v \tag{30}$$

to system (1) to yield an augmented system

$$\dot{\bar{x}} = \begin{bmatrix}\dot{x}\\ \dot{z}\end{bmatrix} = \begin{bmatrix}A & BM_2\\ 0 & J_2\end{bmatrix}\begin{bmatrix}x\\ z\end{bmatrix} + \begin{bmatrix}BN_2\\ L_2\end{bmatrix} v = \bar{A}\bar{x} + \bar{B}v \; ; \; y = [C\ 0]\bar{x} = \bar{C}\bar{x} \tag{31}$$

Our objective in specific mathematical terms is to find the parameters of the compensator such that $\bar{C}_i\bar{A}^k\bar{b}_j \equiv 0, k = 0,1,..(n+\delta-1); i \neq j, i,j = 1,2,...m$ and at least one of the elements of the sequence $\{\bar{C}_i\bar{A}^k\bar{b}_i \neq 0\ k = 0,1,2,..(n+\delta-1)$ is not equal to zero for each i = 1,2,..m.

Let q_i, i = 1,2,..m be the first integers such that

$$C_i A^{q_i} = \sum_{j=1}^{q_i} \varepsilon_{ij} C_i A^{j-1} \; ; \; \sum_{i=1}^{m} q_i = q \tag{32}$$

The method of finding the parameters of the compensator is similar to that presented in the previous section. Define

$$P'_i = [C'_1, A'C'_i, .., A'^{q_i-1}C'_i] \; ; \; P' = [P'_1, P'_2, .., P'_m] \tag{33}$$

and

$$\Gamma' = [\Gamma'_1, \Gamma'_2, ..., \Gamma'_m] \; ; \; \Gamma_i = [P_i, \Lambda_i] \tag{34}$$

where Λ_i is some q_i x (q-n) matrix such that Γ is nonsingular. Similar to lemma 4, a choice of Λ_i such that Γ is nonsingular is

$$\Lambda_i = \begin{bmatrix} & \underline{0} & \\ \underline{0} & I_{q_i-d_i-1} & \underline{0}\end{bmatrix} \; ; \; \Lambda' = [\Lambda'_1, \Lambda'_2, ..., \Lambda'_m] \tag{35}$$

with these definitions, it is seen that

$$PA = \Theta P \tag{36}$$

where

$\Theta = \text{diag}\,(\Theta_1, \Theta_2, ...\Theta_m); \; \Theta_i = \left[\begin{array}{c|c} 0 & I_{q_i-1} \\ \hline \multicolumn{2}{c}{\varepsilon_{i1} ... \varepsilon_{iq_i}} \end{array}\right]$. Now choose

$$\Gamma \begin{bmatrix}BM_2\\ J_2\end{bmatrix} = \Theta\Lambda \text{ and } \Gamma\bar{B} = \Gamma\begin{bmatrix}BN_2\\ L_2\end{bmatrix} = \text{diag}\,(\bar{b}_1, \bar{b}_2, ...\bar{b}_m) \tag{38}$$

where the (q_ix1) vector b_i is given by $b'_i = [0\ 0\ ...\ 1]$. Combining (36) and (38) we get

$$\Gamma\bar{A} = \Theta\Gamma \tag{39}$$

which implies that

$$\bar{C}_i\bar{A}^{q_i} = \sum_{j=1}^{q_i} \varepsilon_{ij}\bar{C}_i(\bar{A})^{j-1} \tag{40}$$

and Γ contains $\bar{C}_i$, $\bar{C}_i(\bar{A})$, ...$\bar{C}_i(\bar{A})^{q_i-1}$ as its rows. Using (38), (34) and (35), it can be established by direct calculation that

$$\bar{C}_i(\bar{A})^j\bar{b}_k \equiv 0; \; i \neq k, \; j = 0,1,...q, \; i,k = 1,2,...,m \tag{41}$$

$$\bar{C}_i(\bar{A})\bar{b}_i \equiv 0; \; j < (q_i-1); \; i = 1,2,..m; \; \bar{C}_i(\bar{A})^{q_i-1}\bar{b}_i = 1 \tag{42}$$

Now from (40)-(42) it can be concluded the system(31)is decoupled. From lemma1, it is also true that (31) is completely controllable and observable.

Equations (38) can be solved for M_2, J_2, N_2 and L_2 in a similar manner as in section 4. The following expressions can be obtained for M_2 and J_2

$$M_2 = \bar{D}^{-1}\,\text{diag}\,\{g_1, g_2, ...g_m\} \tag{43}$$

$$J_2 = P_2\bar{T}^{-1}\bar{Q}_1\Lambda + \bar{\Theta}_2\Lambda \tag{44}$$

where (q_i-d_i-1) row vector g_i is given by $g_i = [1,0,0..0]$ and $P_2' = [(C_1 A^{d_1+1})',..(C_1 A^{q_1-1})',...(C_m A^{q_m-1})']$ The (nxq) matrix $\bar{\Theta}_1$ and the $(d_i+1) \times q_{ii}$ matrix $\bar{\Theta}_{ii}$ are given by $\bar{\Theta}_1 = \mathrm{diag}(\bar{\Theta}_{11},\bar{\Theta}_{22},..\bar{\Theta}_{mm})$; $\bar{\Theta}_{ii} = [\underline{0} \quad I_{d_i+1} \quad \underline{0}]$; $\bar{\Theta}_2$ is the $(q-n)xq$ matrix containing the rows of Θ not included in $\bar{\Theta}_1$. N_2 and L_2 can be solved in a similar way. Thus the following theorem is proved.

Theorem 3: Assume that $d = n$ and $\det D \neq 0$. Then it is possible to compensate system (1) with a compensator of order $(q - n) = \delta$ such that the augmented system is decoupled, controllable and observable.

The decoupled system dynamics are governed by $y_i^{(q_i)} + \sum_{j=1}^{q_i} \varepsilon_{ij} y_i^{(j-1)} = \nu_i$ for $i = 1,2,..m$. If one desires to compensate the decoupled system to achieve desirable transient response and stability properties using measurable variables one can proceed as follows. Observe that the measurable variables are given by $[y' \; z'] = \bar{x}' \, \mathrm{diag}\,\{C',I\}$. Now from equation (39) and (38) it is true that the co-ordinate transformation $\hat{x} = \Gamma\bar{x}$ yields $\Gamma\bar{A}\Gamma^{-1} = \Theta$, $\Gamma\bar{B} = \mathrm{diag}\{\bar{b}_1,\bar{b}_2,...\bar{b}_m\}$ and let $\mathrm{diag}\{C,I\}\Gamma^{-1} = \hat{C}$. It is our aim to find an $mx(m+\delta)$ matrix $\hat{K}$ such that $(\bar{A}+\bar{B}\hat{K}\hat{C})$ has acceptable performance while retaining the noninteraction. From the structure of Θ and $\Gamma\bar{B}$, it can be observed that, to retain the decoupled nature of (31) $\hat{K}\hat{C}$ must be of the form $\hat{K}\hat{C}' = \mathrm{diag}\{\hat{f}_1,\hat{f}_2,...\hat{f}_m\}$ for some $(1xq_i)$ vector $\hat{f}_i$. This equation can be solved easily for $\hat{K}$ and its free parameters may be chosen to achieve acceptable or desirable performance.

Conclusions: This paper has presented a treatment for decoupling a class of systems ($d = n$, $\det D \neq 0$) using output feedback. It may be noted that the viewpoint of generating compensators is different from those presented by Silverman and Payne [4] and Howze and Pearson [5]. It is felt that the approach presented is more nearer to the classical approaches given in [16-18]. It is possible to generalize some of the results of this paper to Include a larger class of systems. For example, Theorem 1 can be generalized easily by making the reference to the ID System of Gilbert [2].

Even though decoupling can be achieved by output feedback, stability is not guaranteed in general. However if a stable decoupled system is obtained then the problem of state estimation via observers is eliminated. Since the decoupled systems obtained via theorems 1,2, and 3 are controllable and observable, it may be possible to obtain the desired performance by using design techniques for single variable systems.

Acknowledgements: This work was partially supported by the National Research Council of Canada under grant No. A1646.

References:

1. P.L. Falb and W.A. Wolovich, "Decoupling in the design of multivariable systems", IEEE Trans. Automat. Control, Vol. AC-12, pp. 651-659, Dec. 1967..
2. E.G. Gilbert, "The decoupling of multivariable systems by state feedback" SIAM J. Control, Vol. 7, pp. 50-64, February 1969.
3. A.S. Morse and W.M. Wonham, "Status of noninteracting control", IEEE Trans. Automat. Control, Vol. AC-16, pp. 568-582, December, 1971.
4. L.M. Silverman and H.J. Payne, "Input-output structure of linear systems with application to the decoupling problem", SIAM J. Control, Vol. 9, No. 2, May 1971, pp. 199-233.
5. J.W. Howze and J.B. Pearson, "Decoupling and arbitrary pole placement in linear systems using output feedback", IEEE Trans. Automat. Control, Vol. AC-15, pp. 660-663, December 1970.
6. J.W. Howze, "Necessary and sufficient conditions for decoupling using output feedback", IEEE Trans. Automat. Contr., Vol. AC-18, pp. 44-46, February 1973.
7. M.J. Denham, "A necessary and sufficient condition for decoupling using

output feedback", IEEE Trans. Automat. Control, Vol. AC-18, pp. 535-536, October 1973.

8. I.H. Muffi, "On the observability of decoupled systems", IEEE Trans. Automat. Control, Vol. AC-14, pp. 75-77, February 1969.
9. ________, "Some results on the decoupling of multivariable systems", Int. J. Control, Vol. 14, No. 3, pp. 477-485, 1971.
10. P.L. Falb and W.A. Wolovich, "On the decoupling of multivariable systems," Preprints, JACC, Philadephia, Pa., June 1967, pp. 791-796.
11. L.M. Silverman, "Inversion of multivariable linear systems", IEEE Trans. Automat. Control, Vol. AC-14, No. 3, pp. 270-276, June 1969.
12. E.M. Cliff and Lunze, F.H., "Application of geometric decoupling theory to synthesis of aircraft lateral control systems", Journal of Aircraft, Vol. 9, No. 11, pp. 770-776, November 1972.
13. C.A. Winsor, "Model referenced control system design", Ph.D. thesis, 1969, Rensselaer Polytechnic Institute, Electrical Engineering.
14. P.D. McMorran,"Design of gas turbine controller using inverse Nyquist Method", Proc. IEE, Vol. 117, No. 10, pp. 2050-2056, October 1970.
15. H.H. Rosenbrock, "State space and multivariable theory", 1970, Thomas Nelson and Sons Ltd., London.
16. J. Peschon,"Disciplines and techniques of system control", New York, Blaisdell, 1965, Ch. 3.
17. J.T. Tou, "Modern control theory", 1964, McGraw-Hill Book Co. N.Y., ch.1
18. R.J. Kavanagh, "Multivariable control system synthesis", AIEE, Applications and Industry, November 1958.

A GRAPH MATRIX APPROACH TO STRUCTURAL CONTROLLABILITY

JONAS BUTVILA AND ROLAND PRIEMER
Information Engineering Department
University of Illinois Chicago Circle
Chicago, Illinois 60680

ABSTRACT

Structural controllability of a system, described by $\dot{x} = Ax+bu$, can be determined by investigating a graph G, derived from (A,b), for certain properties. It is shown that these properties can more readily be determined by treating (A,b) as a connection matrix, and investigating various Boolean determinants of this matrix.

I. INTRODUCTION

A recent paper by Lin [1] defined the concepts of structure and structural controllability of a single input n^{th} order system described by

$$\dot{x} = Ax + bu$$

based on a graph G derived from the pair (A,b). This paper presents a method for determining structural controllability which involves operations on a matrix derived from (A,b) rather than the inspection of G for certain properties.

Two pairs (A,b) and $(\tilde{A},\tilde{b})$ have the same structure, if both pairs have identical zero entries. Zero entries are those which are known exactly, while all other entries are presumed to be known only approximately. A pair (A,b) is by definition [1] structurally controllable if and only if there exists a completely controllable [2] pair (A_o,b_o) which has the same structure as (A,b).

Since only the structure of the system is of interest, the values of the nonzero entries in the system matrices are not relevant, and therefore in the following all matrices have only binary values. The zero locations remain the same, but each of the nonzero values is replaced by a one.

For convenience the nx(n+1) Boolean matrix (A b) will be called the system matrix denoted by S.

The graph G of a pair (A,b) is constructed by considering a graph which contains n+1 nodes. For every nonzero entry s_{ij} of the system matrix S, the graph contains an oriented edge going from node v_j to node v_i. For example, a system described by the following system matrix has the associated graph G shown in Fig. 1.

$$S = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

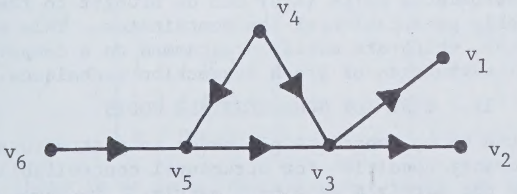


Fig. 1. A graph containing a nonaccessible node and a dilation.

Lin showed that all structurally uncontrollable pairs (A,b) can be brought to one of the following two forms by a permutation of the coordinates. The first form is

$$A = \begin{bmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{bmatrix} \qquad b = \begin{bmatrix} 0 \\ b_2 \end{bmatrix} \tag{1}$$

where $A_{11} \varepsilon R^{kxk}$, $A_{21} \varepsilon R^{(n-k)xk}$, $A_{22} \varepsilon R^{(n-k)x(n-k)}$, $b_2 \varepsilon R^{(n-k)}$, and $1 \leq k < n$. (Lin has $k \leq n$, but if $k=n$ then $A_{11}=A$, which is not a possible condition.) The 0 denotes a matrix or vector containing only zero entries. The second form is the system matrix S written as

$$S = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} \tag{2}$$

where s_1 is a kx(n+1) matrix, $k \geq 1$, with no more than (k-1) nonzero columns (all other columns of s_1 have only zero entries) and s_2 is an (n-k)x(n+1) matrix. A nonzero matrix is a matrix that contains at least one nonzero entry in every row and column, and if any row and column that share a common nonzero element are removed, then the remaining submatrix also contains at least one nonzero entry in every row and column. Nonzero columns are the columns which compose a nonzero matrix, i.e., if there are k nonzero columns in a nxn matrix, then these k columns make up a nxk nonzero matrix.

Systems of type (1) have a nonaccessible node in the graph G [1]. A nonaccessible node is a node v_j which cannot be reached by a directed path from node v_{n+1}. Note that edges leaving the node v_{n+1} (called the origin) are due to nonzero entries in b. Node v_4 of Fig. 1 is a nonaccessible node, since it cannot be reached from node v_6 by a directed path.

Systems of type (2) contain a dilation [1]. A dilation is defined by the existence of a set S_n of k nodes in the vertex set of the graph G (not containing the origin v_{n+1}) such that there are no more than (k-1) nodes v_j in $T(S_n)$. $T(S_n)$ denotes the set of all the nodes v_j with the property that there exists an oriented edge from v_j to a node in S_n. Since nodes v_1 and v_2 of Fig. 1 can be placed in the set S_n, then the set $T(S_n)$ contains v_3, and by definition, the graph G contains a dilation.

A system is structurally uncontrollable if and only if its matrices can be brought to form (1) or (2) [1]. Graphically, this is equivalent to saying either there is a nonaccessible node or a dilation present in G. For complex systems neither of these properties can be easily determined. Lin derived a special graph called a cactus and showed that if any graph G is spanned by a cactus, then neither of the conditions for structural uncontrollability can exist, and therefore the system is structurally controllable. The difficulty with this method is that deriving a graph G and then searching for another spanning subgraph is computationally inefficient. This paper presents an alternative method based on the result that all structurally uncontrollable pairs (A,b) can be brought to the two forms (1) and (2) by a suitable permutation of the coordinates. This method uses only matrix manipulations, which are easily programmed on a computer, thereby eliminating the disadvantages of graph inspection techniques.

II. TEST FOR NONACCESSIBLE NODES

Since there can be no nonaccessible nodes in a structurally controllable system, a necessary condition for structural controllability can be developed by using the matrix S as a path matrix. S, however, is not a square matrix. Since there can be no arcs going from node v_j to v_{n+1}, for completeness, a new (n+1) row is added consisting of (n+1) zeros. For convenience, this matrix is then transposed, resulting in

$$P^{(0)} = \begin{bmatrix} S^T & e \end{bmatrix} \tag{3}$$

where e is a (n+1) column vector containing only zeros. $P^{(0)}$ is a matrix showing all arcs in G and their orientations, such that

$p_{ij}^{(0)} = 1$, if there exists an arc from node v_i to node v_j, and

$p_{ij}^{(0)} = 0$, if there is no arc from v_i to v_j.

All possible paths between nodes can be found by using a simplified version of the Revised Cascade Method [4]. A new matrix $P^{(1)}$ can be generated from $P^{(0)}$ according to the following logic equation.

$$p_{ij}^{(1)} = p_{ij}^{(0)} \text{ OR } \left\{ p_{i1}^{(0)} \text{ AND } p_{1j}^{(0)} \right\} \tag{4}$$

A nonzero entry $p_{ij}^{(1)}$ indicates that there is either a directed arc from v_i to v_j, or that a directed path from v_i to v_j contains v_1 as an intermediate node [4].

This iterative technique continues generating new matrices according to

$$p_{ij}^{(k)} = p_{ij}^{(k-1)} \text{ OR } \left\{ p_{ik}^{(k-1)} \text{ AND } p_{kj}^{(k-1)} \right\}. \tag{5}$$

A nonzero $p_{ij}^{(k)}$ indicates that an arc exists from v_i to v_j or that a path from v_i to v_j exists which contains any subset of the set of nodes v_1 to v_k as intermediate nodes.

This process is terminated when $P^{(n)}$ has been generated. Since there can be no paths with v_{n+1} as an intermediate node, all paths with all possible intermediate nodes have been found. Any nonaccessible nodes can be found by examining row (n+1) of $P^{(n)}$. Any zero element $p_{(n+1)j}^{(n)}$ $(j \neq n+1)$ indicates that there is no directed path from node v_{n+1} to node v_j, and therefore v_j is nonaccessible. To illustrate, this method is applied to the first example, and

$$S = \begin{bmatrix} 0&0&1&0&0&0 \\ 0&0&1&0&0&0 \\ 0&0&0&1&1&0 \\ 0&0&0&0&0&0 \\ 0&0&0&1&0&1 \end{bmatrix}, \quad P^{(0)} = P^{(1)} = P^{(2)} = \begin{bmatrix} 0&0&0&0&0&0 \\ 0&0&0&0&0&0 \\ 1&1&0&0&0&0 \\ 0&0&1&0&1&0 \\ 0&0&1&0&0&0 \\ 0&0&0&0&1&0 \end{bmatrix}$$

$$P^{(3)} = P^{(4)} = \begin{bmatrix} 0&0&0&0&0&0 \\ 0&0&0&0&0&0 \\ 1&1&0&0&0&0 \\ 1&1&1&0&1&0 \\ 1&1&1&0&0&0 \\ 0&0&0&0&1&0 \end{bmatrix}, \quad P^{(5)} = \begin{bmatrix} 0&0&0&0&0&0 \\ 0&0&0&0&0&0 \\ 1&1&0&0&0&0 \\ 1&1&1&0&1&0 \\ 1&1&1&0&0&0 \\ 1&1&1&0&1&0 \end{bmatrix}.$$

Since $p_{64}^{(5)} = 0$, there is no path from v_6 to v_4, which can be seen in Fig. 1, and the system is structurally uncontrollable. As another example, consider the system with the following S matrix and the graph G shown in Fig. 2.

$$S = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

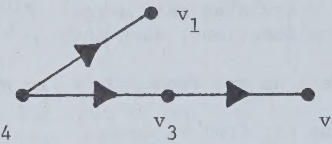


Fig. 2. A graph of a system with a dilation.

The relevant P matrices are

$$P^{(0)} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix}, \qquad P^{(3)} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix}.$$

By inspecting the graph G in Fig. 2 or $P^{(3)}$, there are no nonaccessible nodes. In spite of this, the system is structurally uncontrollable due to a dilation, i.e., $S_n = \{v_1, v_3\}$ and $T(S_n) = \{v_4\}$. Unfortunately, a path matrix contains no information on dilations, and therefore testing $P^{(n)}$ is a necessary but not sufficient condition for structural controllability.

III. DETECTING DILATIONS

The other necessary condition for structural controllability can be resolved by testing for systems of type (2) using Boolean determinants [3].

Since the matrices under investigation are Boolean in nature, it is possible to apply Boolean operations on them. A square Boolean matrix has a Boolean determinant, equal to one or zero, which can be found by replacing numeric operations by logic operations as follows:

Addition, Subtraction → Logical OR (+)

Multiplication → Logical AND (·)

For example,

$$\mathrm{Det}_B \begin{bmatrix} a & b \\ c & d \end{bmatrix} = a \cdot d + b \cdot c \ .$$

As shown in [3], a Boolean determinant can be evaluated by an expansion of minors analogous to the evaluation of ordinary determinants, and the interchange of two rows or two columns does not change the value of the Boolean determinant.

Theorem: The Boolean determinant of an nxn Boolean matrix Q equals zero if and only if Q can be brought to

$$Q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \tag{6}$$

by row interchanges, where q_1 is a kxn matrix with no more than (k-1) nonzero columns ($k \geq 1$) and q_2 is an (n-k)xn matrix.

Proof: There is no loss in generality if after column interchanges Q is assumed to be in the following form:

$$Q = \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix} \tag{7}$$

where q_{11} is a kx(k-1) maximum nonzero matrix, q_{12} is a kx(n+1-k) zero matrix, q_{21} is a (n-k)x(k-1) maximum nonzero matrix, and q_{22} is a (n-k)x(n+1-k) maximum nonzero matrix. A maximum nonzero matrix contains no zero entries. A minimum nonzero matrix contains the minimum number of nonzero entries such that the minimum condition in the definition is met.

Expanding Q by minors yields

$$\mathrm{Det}_B Q = \left|M_{11}\right|_B + \left|M_{12}\right|_B + \;.....\; + \left|M_{1(k-1)}\right|_B. \tag{8}$$

Each M_{ij} in (8) is of the form

$$M_{ij} = \begin{array}{c} \\ (k-1) \\ (n-k) \end{array} \begin{array}{c} \begin{array}{cc} (k-2) & (n+1-k) \end{array} \\ \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \end{array}.$$

Since all minors are identical,

$$\mathrm{Det}_B Q = \left|M_{11}\right|_B.$$

Continuing the evaluation by expansion of minors and simplifying using rules of logic, eventually the following is reached:

$$\mathrm{Det}_B Q = \begin{array}{c} \\ (3) \\ (n-k) \end{array} \left| \begin{array}{c} \begin{array}{cc} (2) & (n+1-k) \end{array} \\ \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \end{array} \right|_B = \begin{array}{c} \\ (2) \\ (n-k) \end{array} \left| \begin{array}{c} \begin{array}{cc} (1) & (n+1-k) \end{array} \\ \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \end{array} \right|_B$$

$$\mathrm{Det}_B Q = \begin{array}{c} \\ (1) \\ (n-k) \end{array} \left| \begin{array}{c} (n+1-k) \\ \begin{bmatrix} 0 \\ 1 \end{bmatrix} \end{array} \right|_B = 0. \tag{9}$$

Changing nonzero entries in q_{11}, q_{21} or q_{22} to zero, generates all matrices of form (6) (column order is ignored, since this does not affect the Boolean determinant), but cannot change the value of the determinant to one. No zeros in q_{12} can be changed to one since this would alter (6). Therefore, all Q matrices with the structure of (6) must have a Boolean determinant equal to zero. Since row interchanges do not change the value of the Boolean determinant, any matrix that can be brought to (6) by row interchanges, also has a Boolean determinant equal to zero.

All matrices Q which cannot be brought to form (6) by row interchanges, can be brought to the form

$$Q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \tag{10}$$

where q_1 is a kxn matrix which contains at least k and at most n nonzero columns, and q_2 is the same as in form (6). Necessity is demonstrated by showing that matrices of form (10) or those which can be brought to form (10) have a Boolean determinant equal to one.

Q is assumed after column interchanges to be in the following form:

$$Q = \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix} \tag{11}$$

where q_{11} is a kxk minimum nonzero matrix, q_{12} is a kx(n-k) zero matrix, q_{21} is a (n-k)xk zero matrix, and q_{22} is a (n-k)x(n-k) minimum nonzero matrix. For simplicity, the nonzero entries of q_{11} and q_{22} are placed on the main diagonal. Since they are both square and diagonal to each other, Q has '1' entries on the main diagonal. Just as for conventional determinants, the value of the Boolean determinant is the logical product of the elements on the main diagonal if all other entries are zero, and therefore the Boolean determinant is one. Again, any matrix that can be brought to form (11) by an interchange of rows and columns will also have a Boolean determinant equal to one.

All matrices of form (10) can be generated (ignoring the column order) by changing zero entries to nonzero entries anywhere in (11). This will not change the system to that described by (6), and this also will not change the value of the determinant to zero. Changing any one in (11) to zero will change the structure to the structure of (6). Any matrix Q that cannot be brought to form (6), can be brought to form (10) and has a Boolean determinant equal to one.

A permutation of coordinates for a system with a system matrix S consists of reordering the variables and affects the matrix S by interchanging the rows and columns to correspond to the new order.

If S can be brought to form (2) by row interchanges, then a permutation of coordinates from the original set to a new set which has the order of the interchanged rows will also bring S to form (2). This results since the order of the columns is immaterial in form (2).

Using the theorem, various Boolean determinants involving the matrix S can be examined and related to the system's structure, thereby determining if a structure of the form (2) exists. This is done by evaluating $\mathrm{Det}_B S_{-i}$ for i=1 to (n+1). S_{-i} is the square matrix derived from S by the removal of the i^{th} column.

If $\mathrm{Det}_B S_{-i} = 1$ for some i, then S_{-i} has the form (10), or it can be brought to (10) by an interchange of rows. If the missing column is added and the columns are reordered to correspond to the new ordering of the rows, then S has undergone a permutation of coordinates and is in a form similar to (2). However, s_1 contains at least k nonzero columns and therefore no dilations exist.

If $\mathrm{Det}_B S_{-i} = 0$ for all i=1 to n+1, then S cannot be brought to the form containing k or more nonzero columns in s_1. Consequently s_1 contains at most (k-1) nonzero columns, and therefore, a dilation exists.

To demonstrate this method, the system of the second example will be checked.

$$\mathrm{Det}_B S_{-1} = \begin{vmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}_B = 0 \qquad \mathrm{Det}_B S_{-2} = \begin{vmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}_B = 0$$

$$\mathrm{Det}_B S_{-3} = \begin{vmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{vmatrix}_B = 0 \qquad \mathrm{Det}_B S_{-4} = \begin{vmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{vmatrix}_B = 0$$

Since all $\mathrm{Det}_B S_{-i} = 0$, a dilation exists and the system is structurally uncontrollable, as was seen earlier.

Another example with S given by

$$S = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 \end{bmatrix}$$

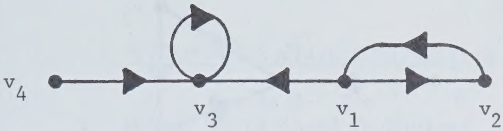


Fig. 3. A graph containing two nonaccessible nodes, but no dilations.

and a graph G as shown in Fig. 3, has the following determinants:

$$\text{Det}_B S_{-1} = 0$$

$$\text{Det}_B S_{-2} = 0$$

$$\text{Det}_B S_{-3} = 1$$

$$\text{Det}_B S_{-4} = 1.$$

Since for i equal to 3 and 4, the determinant is equal to one, there are no dilations. In spite of this, the system is structurally uncontrollable since, as can be seen in the graph, nodes v_1 and v_2 are nonaccessible from v_4.

IV. SUMMARY

Section II of this paper demonstrates a test which determines if a system matrix is of form (1). Section III gives a test for determining system matrices that are of form (2). According to Lin if neither form is present, the system is structurally controllable, hence by performing both tests on the matrix S, the structural controllability of the system can be determined.

As an example the following system is analyzed using both tests.

$$S = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}$$

$$P^{(0)} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad P^{(3)} = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

The fourth row of $P^{(3)}$ has no significant zeros, i.e., there are no nonaccessible nodes.

The Boolean determinants are as follows:

$$\text{Det}_B S_{-1} = 1$$

$$\text{Det}_B S_{-2} = 0$$

$$\text{Det}_B S_{-3} = 0$$

$$\text{Det}_B S_{-4} = 1.$$

Since $\text{Det}_B S_{-i} = 1$ for $i = 1$ and 4, there are no dilations. The two tests show that neither forms (1) or (2) are present, and as can be seen from Fig. 4 the system is structurally controllable.

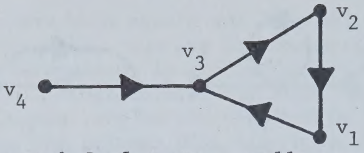


Fig. 4. A graph G of a structurally controllable system.

V. CONCLUSION

This paper presented two necessary conditions for structural controllability, which when combined give a necessary and sufficient condition. This is done without resorting to graphical inspection techniques. By investigating the path matrix $P^{(n)}$ associated with a system matrix S, one cause of structural uncontrollability can be investigated. By calculating the various Boolean determinants the other cause of structural uncontrollability is tested for; therefore the structural controllability of a system can be determined.

The use of Boolean determinants is now under investigation in relation to structural controllability and observability of multi-input/multi-output systems, and will be reported on shortly.

REFERENCES

[1] Lin, C. T., "Structural Controllability," IEEE Trans. Automat. Contr., vol. AC-19, no. 3 pp. 201-208, June 1974.

[2] Chen, C. T., Introduction to Linear System Theory. New York: Holt, Rinehart and Winston, Inc., 1970.

[3] Flegg, H. G., Boolean Algebra and its Application. New York: John Wiley and Sons, Inc., 1964.

[4] Elmaghraby, S. E., Some Network Models in Management Science. Berlin: Springer-Verlag, 1970.

A TOPOLOGY FOR STATE CONSTRUCTION

Norman Y-K Foo
School of Advanced Technology
State University of New York
Binghamton, NY 13901

Abstract

A brief outline of a topology for Nerode State construction is presented. The essence of the procedure is to define an equivalence relation on the product of the input space and the time set, and imposing the quotient topology on the quotient space.

1.1 Introduction

For brevity the elementary notions of systems and automata theory as may be found in ([K1], [A1]) are assumed. Proofs are also omitted, but may be found in [F1] and forthcoming papers.

An input space $\mathcal{U}$ consists of a family of functions mapping a time set T to a value space U. It is usually assumed that $\mathcal{U}$ is closed under translation concatenation and restriction. Similar remarks hold for the output space $\mathcal{Y}$ and its value space V. When a system is time-invariant and all inputs begin from some finite time, the time set may be taken to be R_+ the non-negative reals, assumptions which are made in the sequel. The theory outlined here is a topological extension of the classical Nerode construction in Automata Theory, which can in principle be generalized to arbitrary topologies on $\mathcal{U}$ and R_+. Mention should be made of a differently motivated but somewhat similar development in [O1] which came to the author's attention recently.

This research was supported in part by NSF Grant GJ-29989X to the Logic of Computers Group, Department of Computer and Communications Sciences, The University of Michigan, Ann Arbor, MI 48104, with which the author was associated.

1.2 Input Segment Space

Input segments in systems theory are generalizations of the free semigroup I^* over an alphabet in automata theory. Consider the backward restriction operator E^t which acts on u in $\mathcal{U}$ to yield $u|_{(0,t)}$. An input segment space $\mathcal{U}$ is obtained from $\mathcal{U}$ by the action of all such E^t operators. Concatenation denoted by o is defined in $\mathcal{U}$ in the obvious way, so that $(\mathcal{U},o)$ is a nonoid, with identity Λ the null segment. (Later we also use the forward restriction operator E_t, defined as its name naturally suggests

The segment space $\mathcal{U}$ may be topologized in several ways giving rise to different modes of convergence. It is pertinent to ask what is meant by the assertion that a collection of segments converges to a given segment. To answer this a different way of looking at segments is explicated.

Consider the cartesian product $\hat{\mathcal{U}} \times R_+$. The relation ρ on $\hat{\mathcal{U}} \times R_+$ defined by

$$(\hat{u}_1,t_1)\rho \quad (\hat{u}_2,t_2) \iff t_1 = t_2 \text{ and } \hat{u}_1|_{[0,t_1)} = \hat{u}_2|_{[0,t_2)}$$

is an equivalence relation. There is a bijection between the quotient space $\hat{\mathcal{U}} \times R_+/\rho$ and the segment space $\mathcal{U}$ given by $[(\hat{u},t)]_\rho \to E^t\hat{u}$. That this is indeed a bijection is easily verified. In fact this bijection is a monoid isomorphism between $\hat{\mathcal{U}} \times R_+/\rho$ and $\mathcal{U}$, with the concatenation operation on the first space defined in the obvious way.

Now, $\hat{\mathcal{U}}$ has a function space topology while R_+ has several possible topologies. (For discrete-time systems, R_+ may be replaced by Z, which is given the discrete topology). It is natural to equip $\hat{\mathcal{U}} \times R_+$ with the product topology and hence $\hat{\mathcal{U}} \times R_+/\rho$ with the quotient topology from $\hat{\mathcal{U}} \times R_+$. In this way the segment space is topologized in a natural manner. Note that the inverse image of a segment $[(\hat{u},t)]\rho$ in $\hat{\mathcal{U}} \times R_+$ is a set $\{(\hat{v},t): \hat{v}|_{[0,t)} = \hat{u}|_{[0,t)}\}$

Theorem 1.2.1

The natural projection $p:\hat{\mathcal{U}} \times R_+ \to \hat{\mathcal{U}} \times R_+/\rho$ is both continuous and open.

Theorem 1.2.2

Let $\hat{\mathcal{U}}$ have the topology of uniform convergence and R_+ the discrete topology.

Suppose $u_n \to u$. Then for any sequence $\{(\hat{u}_n,t_n)\}$ such that $(\hat{u}_n,t_n) \varepsilon p^{-1}(u_n)$, and for any $(\hat{u},t)$ with $(\hat{u},t) \varepsilon p^{-1}(u)$, $t_n = t$ eventually and $E^t\hat{u}_n \to E^t\hat{u}$ uniformly.

Conversely, if $\{(\hat{u}_n,t_n)\}$ is a sequence such that $t_n = t$ eventually and $E^t\hat{u}_n \to E^t\hat{u}$ uniformly, then $p(\hat{u}_n,t_n) \to p(\hat{u},t)$.

Theorem 1.2.3

Let $\hat{\mathcal{U}}$ have the topology of pointwise convergence and R_+ the usual topology.

Suppose $u_n \to u$. Then for any sequence $\{(\hat{u}_n,t_n)\}$ such that $(\hat{u}_n,t_n) \varepsilon p^{-1}(u_n)$, and for any $(\hat{u},t)$ with $(\hat{u},t) \varepsilon p^{-1}(u)$, $t_n \to t$ and $E^t\hat{u}_n \to E^t\hat{u}$ pointwise.

Conversely, if $\{(\hat{u}_n,t_n)\}$ is a sequence such that $t_n \to t$ and $E^t\hat{u}_n \to E^t\hat{u}$ pointwise, then $p(\hat{u}_n,t_n) \varepsilon p(\hat{u},t)$.

1.3 $\mathcal{U}$ as a topological monoid

It was earlier noted that $(\mathcal{U},\circ)$ is a monoid. In fact it is a cancellative monoid. When $\mathcal{U}$ has topological structures as given in the previous section it is possible to say more.

Definition 1.3.1

$(S,\circ)$ as a *right(left)* semitopological monoid if S is both a topological space and a monoid, and $\circ:S \times S \to S$ is continuous on the right (left) component. If it is continuous on both right and left it is called semitopological monoid, while if it is jointly continuous on both components it is called a topological monoid.

The following proposition is easy to prove:

Theorem 1.3.1

$(\mathcal{U},\circ)$ is a topological monoid when $\mathcal{U}$ has the topology of uniform convergence and R_+ has the discrete topology.

In the case when $\mathcal{U}$ has the pointwise convergence topology and R_+ the usual topology things are not quite as simple. However the following is true:

Theorem 1.3.2

$(\mathcal{U},\circ)$ is a right semitopological monoid when $\mathcal{U}$ has the topology of pointwise convergence and R_+ has the usual topology.

The distinction between the last two theorems reflects the difference in the power of the two modes of convergence. Without restrictions on $\hat{\mathcal{U}}$ theorem 1.3.2 cannot be strengthened.

In the sequel we shall call the topology in theorem 1.3.1 the u-topology and that in theorem 1.3.2 the p-topology for short.

In a sense the u-topology is rather more fundamental. If the system under consideration is defined on discrete time, hence taking the time set to be the integers Z, it is natural to insist that a sequence of segments converging to a given segment have domains of definition that are eventually equal, i.e., the characteristic of the u-topology. Admittedly, "naturalness" is somewhat subjective, but in attempting to bend the p-topology to yield results of the same power as the u-topology we have to resort to a measure theoretic device.

1.4 Functions on $\mathcal{U}$

Every causal function $\hat{f}:\hat{\mathcal{U}} \to \hat{\mathcal{Y}}$ induces a function $\hat{f}:\mathcal{U} \to \mathcal{Y}$ in the obvious manner, i.e., if $u \in \mathcal{U}$ and $\mathcal{D}(u) = [0,t)$, then $f(u) = E^t\hat{f}(u)$. For simplicity we shall hence forth denote the length of the domain of u by $|u|$, so that in this case $|u| = t$.

It should be clear that for such functions the output is synchronized with the input as a matter of convention, but this can be easily relaxed.

Lemma 1.4.1

f is continuous if $\hat{f}$ is continuous.

We shall have reason to modify $\hat{\mathcal{U}}$ to let it be the space of functions which are identified if they are equal almost everywhere (a.e.).

In such a case segments may be defined over closed or open intervals without affecting any of the previous results but allowing a modification of the function f. For any $u \in \mathcal{U}$, $|u| = t$, the function $\hat{f}$ cannot depend on the value to the point t, so that in fact the function f induced by $\hat{f}$ may in this instance be defined by either $f:\mathcal{U} \to Y$ or $f:\mathcal{U} \to \mathcal{Y}$, recalling that Y is the value space of segments in $\mathcal{Y}$. These are equivalent definitions when $\hat{\mathcal{U}}$ is a quotient space of functions identified if they are equal a.e.

One significance of the identification of functions equal a.e. is that instances of the proof scheme with this provision are the various L_p spaces which pervade continuous systems theory. More to the point is that in this case theorem 1.3.2 may be strengthened to coincide with theorem 1.3.1.

Henceforth when no explicit mention is made of the choice of topologies for $\mathcal{U}$ it is to be understood that if a conclusion is true for the u-topology, it is also true for the modified p-topology.

We can distinguish two system definitions. If $f:\mathcal{U} \to \mathcal{Y}$ is taken as basic we have a Mealy type system: if $f:\mathcal{U} \to Y$ is taken as basic we have a Moore type system.

2.1 Algebraic Theory

We summarize in this section the basic results (see [A1], [K1]) needed to introduce the substance of the topological theory of the next section. The following two definitions correspond to the Mealy and Moore perspectives respectively.

Definition 2.1.1

Given a system $f:\mathcal{U} \to \mathcal{Y}$, the triple (p,Q,λ) is a factorization of f if $p:\mathcal{U} \to Q$, $\lambda:Q \times \mathcal{U} \to \mathcal{Y}$, and $\lambda(p(u),v) = E_{|u|}f(u \circ v)$ for each $p(u) \in R(p)$, $v \in \mathcal{U}$. Q is called the state set. $p(\Lambda)$ is denoted by q_o and called the initial state.

Definition 2.1.2

Given a system $f:\mathcal{U} \to Y$ the triple (p,Q,λ) is a factorization of f if $p:\mathcal{U} \to Q$, $\lambda:Q \to Y$, and the diagram

$$\begin{tikzcd}
U \arrow[rr, "f"] \arrow[dr, "p"'] & & Y \\
& Q \arrow[ur, "\lambda"'] &
\end{tikzcd}$$

commutes. Q is called the state-set.

In this case again $\beta_{q^\circ} = f$.

We shall not take pains to distinguish between Moore and Mealy formulations unless the circumstances warrant it. It will be understood that whenever the p-topology is used for $\mathcal{U}$ we shall be at liberty to use the former.

Canonical factorizations are those which prove to be minimal in the following sense:

Definition 2.2.1

A (p,Q,λ) factorization is reachable if p is surjective and observable if the map $\psi : Q \to y^{U}$ given by $\psi(q) = \beta_q$ is injective. It is canonical if it is both reachable and observable.

A canonical state construction procedure is obtained through the Nerode right congruence defined next.

Let S_τ represent the shift operator with shift τ.

Definition 2.2.2

$u_1, u_2 \varepsilon \mathcal{U}$ are Nerode-equivalent, written $u_1 N u_2$, if

$$S_{-|u_1|} E_{|u_1|} f(u_1 \circ v) = S_{-|u_2|} E_{|u_2|} f(u_2 \circ v)$$

for each $v \varepsilon \mathcal{U}$.

Theorem 2.2.2

N is a right congruence relation on $\mathcal{U}$.

Theorem 2.2.3

The Nerode factorization $(p, \mathcal{U}/N, \lambda)$ of $f : \mathcal{U} \to \mathcal{Y}$ given by $p : \mathcal{U} \to \mathcal{U}/N$, $\lambda : \mathcal{U}/N \times \mathcal{U} \to \mathcal{Y}$, where $p(u) = [u]$ and $\lambda([u], v) = E_{|u|} f(u \circ v)$, is canonical and deterministic.

For the Moore formulation Theorem 2.2.3 may be restated as

Theorem 2.2.4

The Nerode factorization $(p, \mathcal{U}/N, \lambda)$ of $f : \mathcal{U} \to Y$ given by $p : \mathcal{U} \to \mathcal{U}/N$, $\lambda : \mathcal{U}_N \to Y$ where $p(u) = [u]$ and $\lambda([u]) = f(u)$ is canonical and deterministic. In this case the diagram

$$\begin{tikzcd}
U \arrow[rr, "f"] \arrow[dr, "p"'] & & Y \\
& U/N \arrow[ur, "\lambda"'] &
\end{tikzcd}$$

commutes.

The notation $\mathcal{U}/N$ for a quotient class is standard. λ is well-defined because time-invariance allows the identification of all segments related by a shift. The above theorem gives in essence a reduced automaton in the Mealy sense with state transition function

$$\delta : \mathcal{U}/N \times \mathcal{U} \to \mathcal{U}/N$$

$$([u], v) \to [u \circ v]$$

which is well-defined because N is a right congruence.

2.4 Topological Automata

The algebraic theory of factorizations may be reinforced with the introduction of topologies on the state set. Suppose (p, Q, λ) is a factorization of $f:\mathcal{U} \to \mathcal{Y}$. It may be easily shown that there is a right congruence relation M associated with this factorization, so that $\mathcal{U}/M \subset Q$, and $\mathcal{U}/M = Q$ if and only if p is surjective. The natural topology to place on $\mathcal{U}/M$ is the quotient topology through p, that is, a set θ is open in $\mathcal{U}/M$ if $p^{-1}(\theta)$ is open in $\mathcal{U}$. Then $p:\mathcal{U} \to \mathcal{U}/M$ is continuous. Because $\mathcal{U}/M \subset Q$ for $p:\mathcal{U} \to Q$ to be continuous it is sufficient to say that the quotient topology on $\mathcal{U}/M$ is the relative topology of Q restricted to $\mathcal{U}/M$. This will be assumed throughout.

Using the theory developed in the previous sections a theory of topological automata will be presented. For simplicity most of the results depend on the use of the u-topology, but applies to the modified p-topology as well.

The distinction between the Moore and Mealy formulation is displayed here.

Lemma 2.4.1

The diagram

$$\begin{tikzcd}
u & \mathcal{U} \arrow[r, "\mu_z"] \arrow[d, "p"'] & \mathcal{U} \arrow[d, "p"] & u \circ z \\
{[u]_M} & \mathcal{U}/M \arrow[r, "\mu_{z_*}"] & \mathcal{U}/M & {[u \circ z]_M}
\end{tikzcd}$$

commutes and μ_{z_*} is continuous. Consequently $\forall_z \varepsilon \mathcal{U}\ [u_n] \to [u] \Rightarrow [u_n \circ z] \to [u \circ z]$.

In the terminology of factorizations it is seen that the mapping μ_{z_*} acts as follows:

$$\mu_{z_*}([u]) = \delta([u],z)$$

Thus, the above lemma is in fact a statement that $\delta: \mathcal{U}/M \times \mathcal{U} \to \mathcal{U}/M$ is continuous in $\mathcal{U}/M$.

Lemma 2.4.2

$[u_1] = [u_2] \Rightarrow [u_1 \cdot z] = [u_2 \cdot z]$ for all $z \varepsilon \mathcal{U}$.

Lemma 2.4.3

Let $z_n \to z$. Then $[u \cdot z_n] \to [u \cdot z]$ and is independent of the class representative u.

Combining lemmas 2.4.1 and 2.4.3 leads to the following result.

Theorem 2.4.1

In any factorization $(p, \mathcal{U}/M, \lambda)$ of a system f the state transition function $\delta: \mathcal{U}/M \times \mathcal{U} \to \mathcal{U}/M$ is continuous in each component.

This theorem is strengthened when p is open.

Theorem 2.4.2

If p is open, $\delta:\mathcal{U}/M \times \mathcal{U} \to \mathcal{U}/M$ is continuous.

Theorem 2.4.3

In the Mealy formulation the output function $\lambda:Q \times \mathcal{U} \to \mathcal{Y}$ is continuous in each component if and only if $f:\mathcal{U} \to \mathcal{Y}$ is continuous.

Theorem 2.4.4

If p is open, $\lambda:\mathcal{U}/M \times \mathcal{U} \to \mathcal{Y}$ is continuous if and only if f is continuous.

References

[A1] Arbib, M. A., "Decomposition Theory for Automata and Biological Systems" JACC, August 1971, Systems Structure, IEEE Pub. No. 71C61-CSS.

[F1] Foo, N. Y., "Homomorphic Simplification of Systems" Ph.D. Thesis, 1974, University of Michigan, Ann Arbor.

[K1] Kalman, R. E., Falb, P. and Arbib, M. A., "Topics in Mathematical Systems Theory", McGraw-Hill, 1969.

[O1] Orat, E. T. and Geary, J. A., "Representation of a class of nonlinear systems", Proceedings of the First International Symposium on Category Theory Applied to Computation and Control, 1974, University of Massachusetts, Amherst.

On a Systematic Design of Nonlinear Time-Delay Processes.

M. Jamshidi
Associate Professor of Electrical Engineering
Director of Computing Center
Pahlavi University
Shiraz, Iran.

Abstract

A new approach to the sub-optimal control of nonlinear time-delay systems is presented in this paper. It consists of three stages. In the first stage the original time-delay system state equation is expanded about its nominal functions. The resulting linear time-varrying system with delay is then reduced to a linear non-delay nonhomogeneous system whose optimal control is the same. The third stage is the application of the maximum principle to the transformed system leading to the desired sub-optimal control. The method is applied to a nonlinear system.

INTRODUCTION

The systems with time-delay appear in many industrial control processes. Rolling mills and chemical reactors are only two such plants whose model are accurately represented by nonlinear differential equations with time-delay. The application of the maximum principle to such systems. results in two-point boundary-value problems in which both the delay and advance arguments are present, [1] . In fact, it is the presence of such two-point boundary-value problems which makes any computational procedure impossible. For linear time-delay systems with quadratic cost functional, [2] has developed anylytical expressions for control and performance index. This approach is refined further by [3] which involves the solution of a set of successively coupled partial differential equations. Although such methods avoid the solution of any two-point boundary-value problems, its computational implementation is rather cumbersome and difficult. Another basic method in optimal control of time-delay systems has been the sensitivity approach [4] in which the state, costate and control are expanded in terms of the delay itself and the series coefficients are obtained by nondelay systems computations only. The parameter imbedding approach to the solution of linear time-delay systems proposed by [5] forms a family of problems one of which is not delayed while the last member of the class corresponds with the original problem. This concept is developed to the coupling between linear and nonlinear systems by[7]

A third general approach of optimal control of linear time-delay systems is the transformation approach first introduced by [8] and independently proposed by [9] . The idea behind this approach is the transformation of the original linear time-delay system into a system of non-delay and nonhomogeneous with the same optimal control and cost functional.

This method seems to be very power ful and surprisingly has not received a great deal of attention in the liteature.

In this paper the methods of series expansion and transformation are used to obtain a sub-optimal control for nonlinear time-delay systems. The proposed method consists of three stages. The first stage is a Taylor series expansion of the originally nonlinear system state equation about the nominal state and control functions. The resulting linear time-delay system with time-varrying coefficients is them transformed into a linear non-delay , nonhomogeneous system using the basic approach of 10 . The third stage is the application of the maxinum principle to the transformed system leading to the desired sub-optimal control. This control can then be applied to the original time-delay system to obtain the state.

SUB-OPTIMAL DESIGN

Consider a nonlinear time-delay control process expressed by the following state equation,

$$\dot{X}=F(X,X(t-\tau),\ U,U(t-\tau)) \qquad (1)$$

with initial functions,

$$X(t) = X_0(t)$$

$$U(t) = U_0(t). \qquad -\tau < t < 0 \qquad (2)$$

where X and U are n-dimensional state and m-dimensional control vectors, respectively, is the delay and F(.) is assummed to be a continuously differentiable function of its arguments. The optimal control problem is formulated as follows. It is required to find an optimal control vector U* which satisfies (1) and (2) while minimizing an energy cost functional,

$$J = Q(X(T)) + \int_0^T G(U(t),t)dt. \qquad (3)$$

As it was mentioned earlier, the application of the maximum principle to the above problem requires the solution of a two-point boundary-value problem with both delay and advance terms which is computationally impossible.

The first stage of the proposed technique is the expansion of equation (1) about a nominal trajectory (X_n,U_n). Let (x,u) be the trajectories of deviations between (X,U) and (X_n,U_n), i.e.

$$\begin{aligned} x&=X-X_n \\ u&=U-U_n \end{aligned} \qquad (4)$$

It is assummed that pair (X_n,U_n) satisfy (1), i.e.

$$\dot{X}_n=F(X_n,X_n(t-\tau),U_n,U_n(t-\tau))=F_n \qquad (5)$$

The Taylor series expansion of (1) about the nominal trafectories will be,

$$\dot{X}=F_n+(\partial F/\partial X_n)x+(\partial F/\partial X_n(t-\tau))x(t-\tau)$$

$$+(\partial F/\partial U_n)u+(\partial F/\partial U_n(t-\tau)u(t-\tau) \quad (6)$$

Considering the definitions(4) and assumption(5), equation (6) can be rewritten as,

$$\dot{X}=A(t)X+B(t)X(t-\tau)+C(t)u(t)+D(t)u(t-\tau) \qquad (7)$$

with initial trajectories,

$$X(t)=X_0(t)-X_n(t) \qquad -\tau<t<0 \qquad (8)$$

$$u(t)=U_0(t)-U_n(t)$$

The following theorem provides reduction of the linear time-delay system (7) to ordinary one.

Theorem Dynamical systems I and II, given below, have the same optimal control with respect to the following cost functional,

$$J = q(x(T))+\int_0^T g(u(t),(t))dt. \qquad (9)$$

System I:

$$\dot{X}(t)=A(t)x(t)+B(t)x(t-\tau)+C(t)u(t)+D(t)u(t-\tau) \quad 0<t<T \qquad (9a)$$

$$X(t)=n \qquad -\tau<t<0 \qquad (9b)$$
$$u(t)=$$

System II:

$$\dot{z}(t)=K(t)u(t) \qquad 0<t<T \qquad (10a)$$

$$z(0)=L(0)x(0)+\int_\tau^0 L(t+\tau)B(t+\tau)x(t)dt+$$
$$\int_\tau^0 L(t+\tau)D(t+\tau)u(t)dt \qquad (10b)$$

where the nxn adjoint matrix L is obtained from,

$$\dot{L}(t)=-L(t)A(t)-L(t+\ )B(t+\ ) \qquad t\leq T \quad (11a)$$
$$L(T)=I \qquad (11b)$$
$$L(t)=0 \qquad t>T \quad (11c)$$

and

$$K(t)=L(t)C(t)+L(t+\tau)D(t+\tau) \qquad (12)$$

The proof of this theorem, with some minor modifications, has been considered by Bate [8] . This is essentially the second stage of the design.

The n-demensional vector z, called pseudo-state, can be expressed as,

$$z(t)=L(t)x(t)+\int_{t-\tau}^{t} L(\sigma+\tau)B(\sigma+\tau)x(\sigma)d\sigma + \int_{t-\tau}^{t} L(\sigma+\tau)D(\sigma+\tau)u(\sigma)d\sigma \qquad (13)$$

From equations (11) and (13), it follows, Slatter and Wells [9] that,

$$z(T) = x(T) \qquad (14)$$

The ordinary optimal control problem given by equations (8), (10), (11) and (14) can now be formulated using the necessary conditions of the maximum principle. Define the Hamiltonian function,

$$H(z,p,u,t)=-g(u,t)+p'K(t)u \qquad (15)$$

where p is the costate vector of z, whose equation is obtained

form,

$$\dot{p}=-\partial H/\partial z=0 \ , \ p(T)=q_z(z(T)),. \qquad (16)$$

Hence, the costate P is,

$$p(t)=q_z(z(T))=\text{constant} \qquad (17)$$

The optimality condition of the Hamiltonian H is given as,

$$0=\partial H/\partial u=-g_u(u,t)+K'p(t)=-g_u(u,t)+K'(t)q_z(z(T)) \qquad (18)$$

From the above equation, an open loop expression for the optimal control can be obtained.

For a quadratic cost functional $g(u,t)=\frac{1}{2}u'Ru$, and $q(z(T))=z'(T)Mz(T)$, equation (18) will lead to the optimal closed loop control, 10

$$U^{*}(t)=-R^{-1}K'(t)p(t)z(t) \qquad (19)$$

where P(t) satisfies the following Riccati-type equation,

$$\dot{P}(t)=P(t)K(t)R^{-1}K'(t)p(t) \qquad (20a)$$

$$P(T)=M \qquad (20b)$$

substituting (13) into (19) yields the proposed sub-Optimal designed control vector,

$$u^{*}(t)=-K_1(t)x(t)-\int_{t-\tau}^{t} K_2(t,\sigma)x(\sigma)d\sigma - \int_{t-}^{t} K_3(t,\sigma)u(\sigma)d\sigma \qquad (21)$$

where,

$$K_1(t)=R^{-1}K'(t)P(t)L(t) \qquad (22a)$$

$$K_2(t,\sigma)=R^{-1}K'(t)P(t)L(\sigma+\tau)B(\sigma+\tau) \qquad (22b)$$

$$K_3(t,\sigma)=R^{-1}K'(t)P(t)L(\sigma+\tau)D(\sigma+\tau) \qquad (22c)$$

It is noted that the control law (21) has a direct feedback portion and two on-line portions in $x(t)$ and $u(t)$, delayed during second preceeding the instant t Figures 1 and 2 present, the proposed sub-optimal control design using the linearized and original models, respectively.

COMPUTATIONAL PROCEDURE

In this portion, the computational steps are given. This will be followed by a discussion on the evaluation of adjoint matrix $L(t)$.

Computational Steps

The proposed design method for quadratic cost functional is summarized by the following steps:

1. Integrate equation (11) backward in time and store $L(t)$, $-\tau < t < T$.
2. Compute matrix $K(t)$ using (12).
3. Integrate equation (20) and store the Riccati matrix $P(t)$.
4. For instant t and dummy argument obtain the feedback matrices $K_1(t)$, $K_2(t,\sigma)$ and $K_3(t,\sigma)$ and store.
5. Integrate the terms $K_2(t,\sigma)x(\sigma)$ and $K_3(t,\sigma)u(\sigma)$ in the interval $t-\tau$ and t using (21) to obtain $u^*(t)$.
6. Set $t=t+\Delta t=t+mt$
7. If $t<T$ repeat step 4.
8. $U^*(t)=u^*(t)+U_n(t)$.

It is noted that the above procedure assummes that the step size t is an integral multiple of delay t, i.e. $\Delta t=mt$, $m>1$.

b. Evaluation of Adjoint Matrix L(t)

The evaluation of adjoint matrix $L(t)$ in (11) requires some thought, especially for stable matrices A and B. This equation can be integrated in two basic ways depending on the nature of matrices A,B,C and D. (1) For constant matrices A,B,C, and D, the equation (11) due to Bate (1969) can be 8 changed to,

$$\frac{d}{d\alpha} L_n(\alpha) = L_n(\alpha)A+L_{n-1}(\alpha)B \qquad (23)$$

$$L_n(\alpha) \doteq L_n(T-nt-\alpha) \qquad (24a)$$

$$L_{-1}(\alpha)=0 \qquad (24b)$$

$$L_0(0)=I \qquad (24c)$$

$$L_n(0)=L_{n-1}(\tau) \qquad (24d)$$

where $L_n(\alpha)=L(T-n-\alpha)$

This sequence of non-homogeneous differential equations can be solved, Bate (1969), as follows,

$$L_n(\alpha) = [L_{n-1}(\tau)+\int_0^\alpha L_{n-1}(\alpha')Be^{-A\alpha'}d\alpha']\ e^{A\alpha} \qquad (25)$$

(ii) For time-varying matrices A(t), B(t), C(t), and D(t), the most reasonable method is the mumerical integration of the differential-difference equation (11). This method could have been applied for the case where all the matrices were constant, but the method introduced above is more convenient.

NUMERICAL EXAMPLE

The design method outlined in the previous sections, is applied to a numerical example. Consider a first-order non-linear time-delay system,

$$\dot{X}=-2x^2+XX(t-0.1)+UU(t-0.1) \qquad (26)$$

with initial functions,

$$X_0(t)=10(t+0.1)$$

$$U_0(t)=1. \qquad -0.1\leq t\leq 0 \qquad (27)$$

and the cost functional,

$$J=\tfrac{1}{2}(X(1)-1)^2+\tfrac{1}{2}\int_0^1 (U(t)-1)^2dt \qquad (28)$$

Let the nominal trajectories be $(X_n,U_n)=(1,1)$. The linearized model for the system (26) using the above nominal trajectories become,

$$\dot{X}=-3x+x(t-0.1)+u+u(t-0.1) \qquad (29)$$

$$X(t)=10t$$

$$u(t)=0 \qquad -0.1\leqq t\leqq 0.0 \qquad (30)$$

This completes the first step of the design. Following the computational steps outlined in Section 3, the adjoint variable $\ell(t)$ is obtained by solving,

$$\dot{\ell}(t)=3\ell\quad(t)-\ell(t+0.1)$$
$$\ell(1)=1 \qquad (31)$$

which, according to the procedure outlined, is reduced to the following sequence of differential equations,

$$\dot{\ell}_n(t)=-3\ell_n(t)+\ell_{n-1}(t) \qquad (32)$$

$$\ell_{-1}(t)=0,\ \ell_0(0)=1,\ \ell_{n-1}(0.1).$$

Solving this series of differential equations leads to a value $(t)=\exp(3(t-1))$. For $0 \quad t \quad 1$. The value of $k(t)$ defined in (12), can be obtained as $k(t)=\exp(3(t-1))+\exp(3(t-0.9))$. The second stage privides the transformation of (29) into an ordinary differential equation,

$$\dot{z}=k(t)u(t) \qquad (33)$$

with

$$z(0)=\quad(0)x(0)+\qquad(t+0.1)x(t)dt$$

and cost functional,

$$J=\tfrac{1}{2}\,x^2(1)+\tfrac{1}{2}\int_0^1 u^2(t)dt. \qquad (34)$$

For the non-delay optimal control problem given by equations (33)-(34), the application of the linear regulator theory leads to the following equations,

$$\dot{p}(t)=0 \quad , \quad p(1)=z(1) \qquad (35a)$$

$$p(t)=-p(t)z(t) \qquad (35b)$$

$$u(t)=-K(t)P(t)z(t) \qquad (36)$$

$$\dot{z}(t)=-K^2(t)P(t)z(t) \qquad (37)$$

where **p** is the constant costate and P is the solution to Riccati equation,

$$\dot{P}(t)=-k^2(t)P^2(t), \quad P(1)=1. \qquad (38)$$

This problem was simulated on an IBM -370/135 digital computer. The Riccati variable P(t) and pseudo-state z(t) are shown in Figures 3 and 4. Using the optimal control obtained in (36) and state equation (29) the state X(t) and control U(t) are shown in Figures 5 and 6. The feedback law outlined in (21)-(22), and the optimal closed-loop control can be similarly obtained.

CONCLUSIONS

A three stage design of nonlinear time-delay processes is presented in this paper. The proposed method is based on a Taylor series expansion of the original nonlinear process about a nominal pair of state and control vectors in the first stage. The second stage is the transformation of the resulting time-varying linear system with delay into an ordinary non-delay process. The application of the optimal control theory to the final system and relating it back to the original system constitutes the third stage.

As compared with other methods, the present one seems to be a very convenient one to apply in which comparatively less approximation is involved. The only shortcoming seems to be its limitations to the cost functions involving control varionly. Thus, a logical extension to the method would be to use cost functions other than those involued with minimum energy on control effort.

ACKNOWLEDGEMENT

The author wishes to thank Pahlavi University Computing Center for providing their facilities in simulating the numerical example.

REFERENCES

1. Kharatishvili, G.L. (1961), "The maximum Principle in the Theory of Optimal Processes with Time-Lags", Dokl. Akad. Nauk USSR. 136, 39

2. Krasovskii, N.N. (1963), "Optimal Processes in Systems with Time-Lags," Proc. Second IFAC Congress, pp.327-332, Basel, Switzerland

3. Eller, D.H., J.K. Aggarwal, and H.T. Banks. (1969), "Optimal Control of Linear Time-Delay Systems," IEEE Trans. AC AC-14, 678.

4. Inoue , K.,A. Akashi, K. Ogino, and Y. Sawaragi (1971), "Sensitivity Approaches to Optimization of Linear with Time Delay, IFAC Automatica, 7, 671.

5. Jamshidi, M. and M. Malek-Zavarei (1972), "Suboptimal Design of Linear Control Systems with Time Delay," Proc. IEE, 119, 1743.

6. Chan, H.C. and W.R. Perkins (1973), "Optimization of Time Delay Systems Using Parameter Imbedding," IFAC Automatica, 9, 257

7. Jamshidi, M. (1973) "Suboptimal Control of Coupled Time-Delay Systems," Int. J. Control 17, 995.

8. Bate, R. (1969), "The Optimal Control of Systems with Time Lags," in Advances in Control Systems, 7(C.T. Leondes, Ed.) pp. 177-224, Academic Press, New York

9. Slater, G.L. and W.R. Wells (1972), "On the Reduction of Optimal Time-Delay Systems to Ordinary Ones," IEEE Trans. AC, AC-17, 154.

10. Kalman, R.E. (1960), "Contribution to the theory of Optimal Control." Bol. Soc. Math. Mexicana, 102.

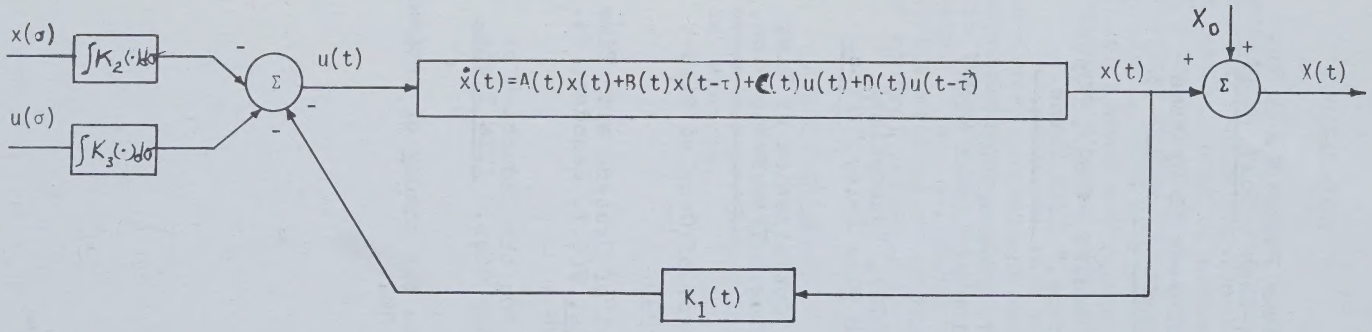

x(σ)
∫K₂(·)dσ
u(σ)
∫K₃(·)dσ
Σ
u(t)
ẋ(t)=A(t)x(t)+B(t)x(t-τ)+C(t)u(t)+D(t)u(t-τ)
x(t)
X₀
Σ
X(t)
K₁(t)

Fig. 1 Proposed Sub-Optimal Control Design Using Linearized Model

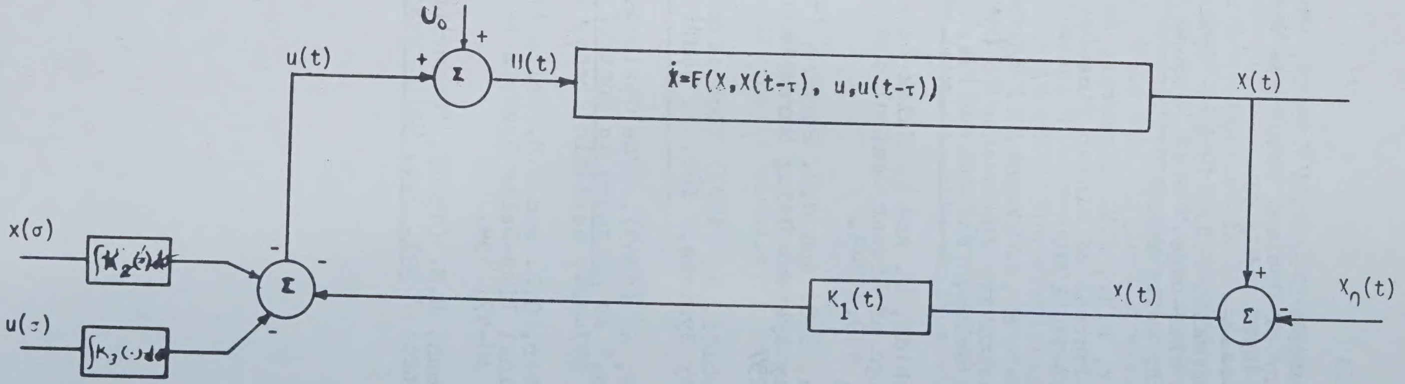

U₀
u(t)
Σ
U(t)
Ẋ=F(X,X(t-τ), u,u(t-τ))
X(t)
x(σ)
∫K₂(·)dσ
u(σ)
∫K₃(·)dσ
Σ
K₁(t)
x(t)
Σ
Xη(t)

Fig. 2 Proposed Sub-Optimal Control Design Using Original Model

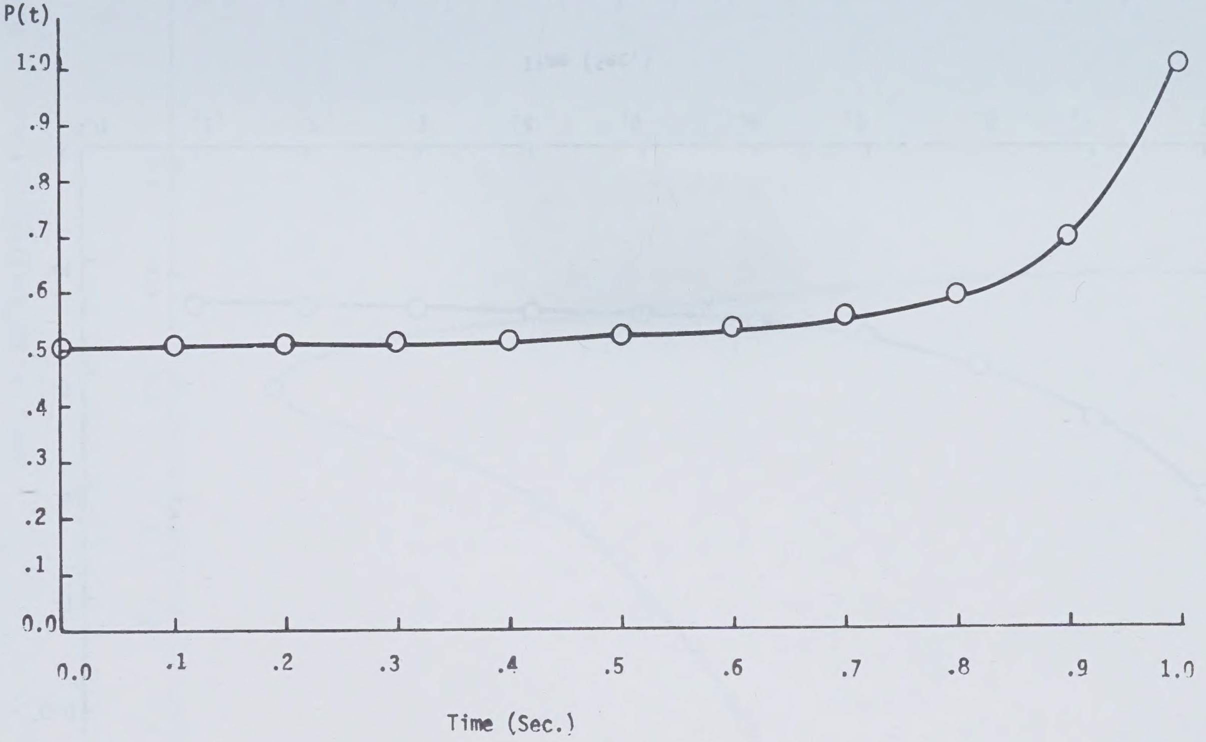


Fig. 3 The Riccati Variable p(t) vs. Time

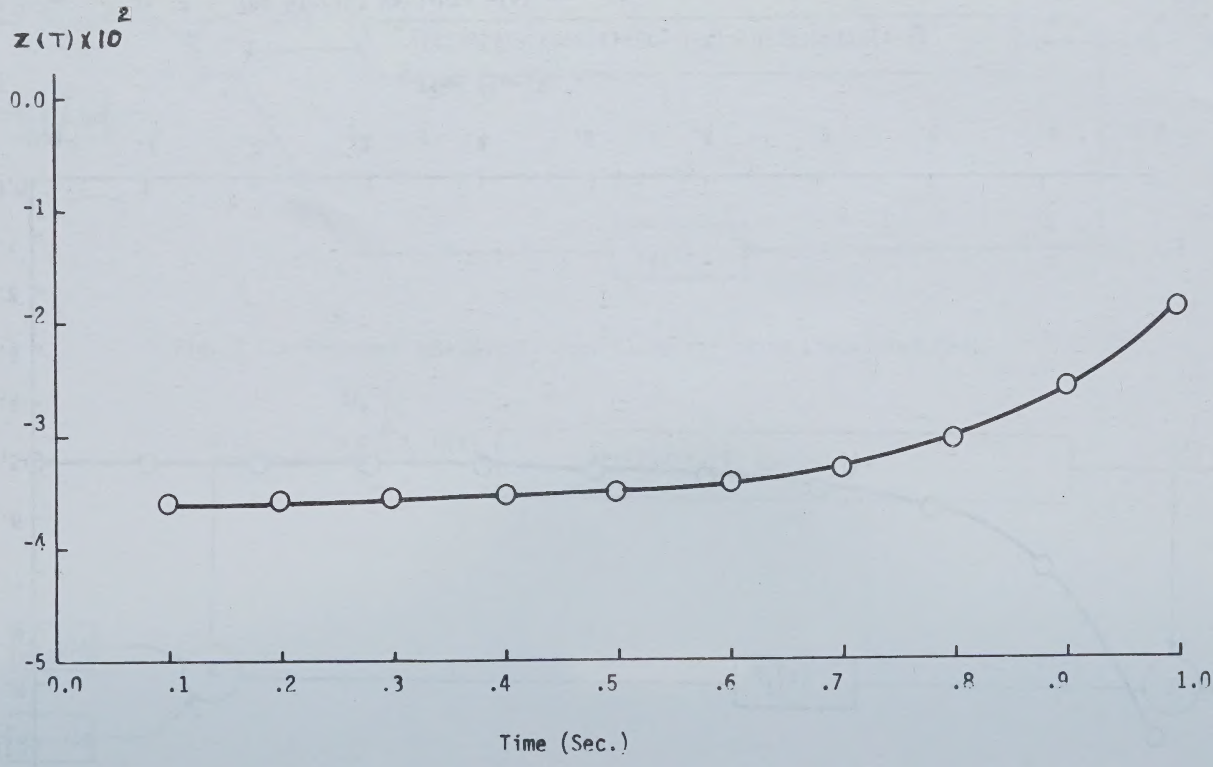


Fig. 4 The Pseudo-State z(t) vs. Time

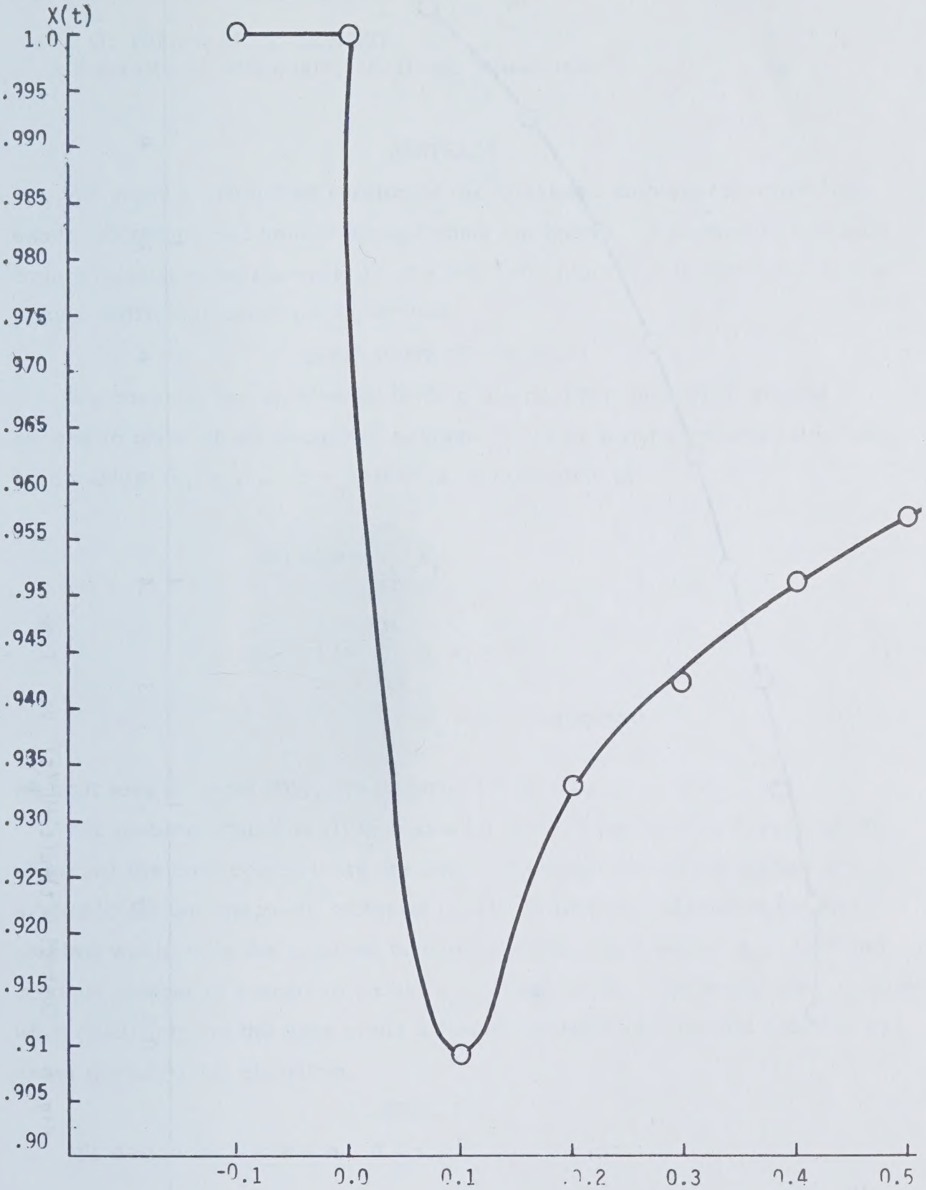


Fig. 5 Stale $X(t)=X(t)+X_n(t)$ vs. Time

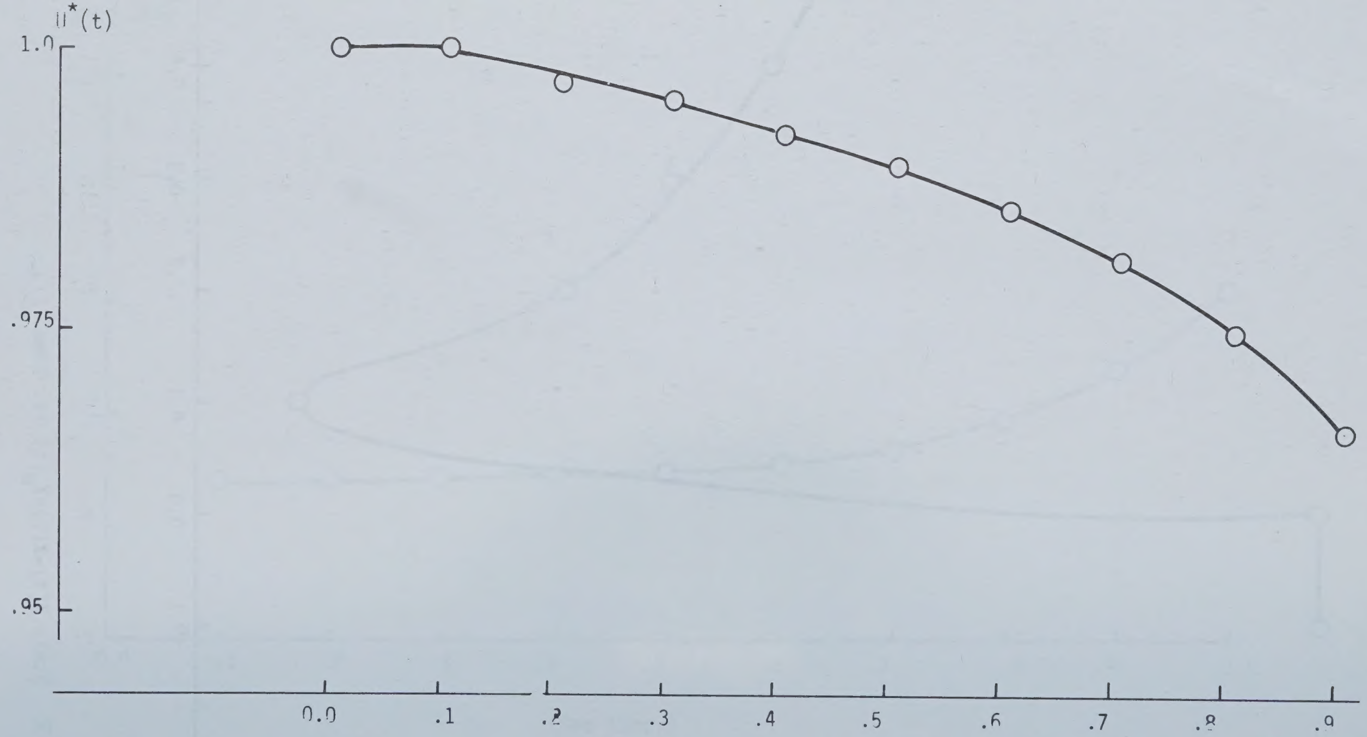


Fig. 6 Control $U(t)=u(t)+U_n(t)$ vs. Time

A STUDY OF A HEURISTIC ALGORITHM [1, 2]

T. C. HU and M. L. LENARD
University of Wisconsin, Madison, Wisconsin

ABSTRACT

We study a simplified version of the knapsack problem for which both exact algorithms and heuristic algorithms are known. A necessary and sufficient condition for the validity of a heuristic algorithm is derived. Also a simple sufficient condition is derived.

STATEMENT OF PROBLEM

We consider the problem of finding the minimum number of stamps needed to make up an amount of postage b when stamps are available with face values $a_1, a_2, \ldots, a_n$; that is, the problem is

$$\text{minimize} \sum_{i=1}^{n} x_i$$

$$\text{subject to} \sum_{i=1}^{n} a_i x_i = b \qquad (1)$$

$$x_i \geq 0, \quad \text{integers.}$$

Without loss of generality, we assume $1 = a_1 \leq a_2 \leq \ldots \leq a_n$.

The problem stated in (1) is a special case of the knapsack problem in which all the cost coefficients are one. Of course, exact algorithms are available for the knapsack problems [1, 2]. A heuristic algorithm for this problem would take the maximal number of stamps with value a_n, then the maximal number of stamps of value a_{n-1}, and so on. We would like to know what conditions on the data would allow us to obtain an optimal solution by using the heuristic algorithm.

RESULTS

We define for $1 \leq k \leq n$, $0 \leq y \leq b$,

---

[1] Sponsored by the United States Army under Contract No. DA-31-124-ARO-D-462.

[2] The material presented here is a summary of results contained in a paper "A study of a heuristic algorithm for knapsack problems of a special type" by T. C. Hu and M. L. Lenard, to appear in SIAM J. on Computing.

$$G_k(y) = \min \sum_{i=1}^{k} x_i$$

$$\text{subject to } \sum_{i=1}^{k} a_i x_i = y \ .$$

Thus, $G_k(y)$ represents the minimum number of stamps of the first k kinds needed to make up an amount of postage y .

Next define $H_k(y)$ to be the number of stamps required by the heuristic algorithm when the amount of postage is y and the first k kinds of stamps are used. Thus,

$$H_{k+1}(y) = \lfloor y/a_{k+1} \rfloor + H_k[y(\bmod a_{k+1})] \ .$$

It is clear that $H_1(y) = G_1(y)$ and $H_2(y) = G_2(y)$ for all y for $a_1 = 1$ and for any value of a_2 .

The first theorem gives necessary and sufficient conditions for establishing the optimality of the heuristic for $k + 1$ stamps given that it is optimal for k stamps.

Theorem 1. Given that $H_i(y) = G_i(y)$ for all y and $i \leq k$. Then $H_{k+1}(y) = G_{k+1}(y)$ for all y if and only if

$$p \geq 1 + H_k(\delta) \ ,$$

where p and δ are defined uniquely by

$$a_{k+1} = p a_k - \delta, \quad 0 \leq \delta < a_k, \ p \geq 2 \ . \tag{2}$$

Thus, if the heuristic algorithm works for k stamps, we can determine whether it will work with one additional stamp of value a_{k+1} by solving a single knapsack problem using the heuristic algorithm, that is, by evaluating $H_k(\delta)$.

Clearly $H_k(\delta) \leq \delta$ which leads us to the simpler sufficient condition given in the following theorem. (This result has also been stated by Kernighan and Johnson [3]).

Theorem 2. Given $H_i(y) = G_i(y)$ for $i = 1, \ldots, k$ and for all y . If the value of δ defined by (2) satisfies

$$0 \leq \delta \leq p-1 .$$

then $H_{k+1}(y) = G_{k+1}(y)$ for all y .

ACKNOWLEDGEMENT

A more general result of the type given in Theorem 1 was first obtained in [4] using a somewhat different argument. The authors wish to express their gratitude for private communication of this work.

REFERENCES

[1] R. E. Bellman and S. E. Dreyfus, "Applied Dynamic Programming", Princeton Univ. Press, Princeton, N. J. 1962.

[2] P. C. Gilmore and R. E. Gomory, "The Theory of Knapsack Functions", J. of ORSA 14 (6), pp. 1045-1074 (1966).

[3] S. C. Johnson and B. W. Kernighan, "Making Change with a Minimum Number of Coins," Bell Telephone Laboratories, Murray Hill, New Jersey, Internal Technical Memorandum, 1971.

[4] M. Magazine, G. L. Nemhauser and L. E. Trotter, Jr., "When the Greedy Solution Solves a Class of Knapsack Problems", Technical Summary Report No. 1421, Mathematics Research Center, University of Wisconsin, Madison, 1974.

INTEGER AND MIXED-INTEGER PROGRAMMING MODELS: GENERAL PROPERTIES[*]

R. R. Meyer
Mathematics Research Center, University of Wisconsin-Madison
Madison, Wisconsin 53706

It is well-known that the fixed charge function,

$$f(x) = \begin{cases} 0 & \text{for} \quad x = 0 \\ 1 & \text{for} \quad x \in (0, b] \end{cases}$$

where b is a given positive constant, may be "represented" by the minimum value of the integer variable u subject to the constraints $0 \leq x \leq bu \leq b$. Such representations are commonly used to formulate minimum-cost facilities-location problems ([2], [3]) as mixed integer programming (MIP) problems.

In this paper, a rigorous definition of a mixed-integer "model" (MIMM) of a function of n real variables is formulated, and the utility of such models is established. The paper also characterizes the classes of functions that can be modelled by three classes of MIMM's, namely, those in which the integer variables must lie in some bounded set, those for which the constraint coefficients are rational, and those not in the previous two classes. It is shown that a MIMM representation is possible when the function to be represented is lower semi-continuous and piecewise-linear, and has a bounded domain. For unbounded domains, sufficient conditions as well as necessary conditions for representability are also described (in the case of functions of a single real variable, conditions that are both necessary and sufficient are presented in [5].) One interesting consequence of these results is that the uncapacitated fixed-charge function (in which the interval (0, b] above is replaced by (0, ∞)) belongs to a class of functions that can be modelled only by resorting to irrational coefficients and unbounded integer variables. Thus, problems in which such cost functions appear cannot be solved through the use of general-purpose MIP codes.

---

[*] This is an extended abstract of a paper to appear in the Journal of Optimization Theory and Applications.

Sponsored by the United States Army under Contract No. DA-31-124-ARO-D-462.

REFERENCES

1. George B. Dantzig, Jon Falkman, and Norman Shapiro, "On the Continuity of the Minimum Set of a Continuous Function", J. Math. Anal. Appl. 17 (1967), pp. 519-548.

2. Robert S. Garfinkel and George L. Nemhauser, Integer Programming, John Wiley, New York, 1972.

3. T. C. Hu, Integer Programming and Network Flows, Addison-Wesley, Reading, Mass., 1969.

4. R. R. Meyer, "On the Existence of Optimal Solutions to Integer and Mixed-Integer Programming Problems", Mathematics Research Center Report #1351, Madison, Wisconsin, 1973.

5. R. R. Meyer, "Mixed Integer Minimization Models for Piecewise-Linear Functions of a Single Variable", Mathematics Research Center Report #1460, Madison, Wisconsin, 1974.

AN EFFICIENT ALGORITHM FOR MAXIMUM K-MATCHING IN WEIGHTED GRAPHS

LEE J. WHITE
Department of Computer & Information Science, Ohio State University,
Columbus, Ohio 43210

This paper considers the problem of obtaining a maximum weight sum matching for a given edge-weighted graph G(V,E). An efficient algorithm is presented which obtains a maximum matching of cardinality k as k sweeps through all feasible values. The weights of these matchings are shown to be concave with parameter k. The computational order of the algorithm is found to be $|V|^2|E|$, and computational experience is cited.

1. INTRODUCTION

Consider a finite graph G(V,E), where V and E denote the set of vertices and edges respectively, and a weighted graph is denoted [G,c], where c_i is a real number associated with edge $e_i \varepsilon E$. For standard definitions used in this paper, see Busacker and Saaty [5]. The incidence matrix A of a graph is defined as

$$a_{ij} = \begin{cases} 1, \text{ if edge } e_j \text{ is incident to vertex } v_i, \\ 0, \text{ otherwise.} \end{cases}$$

The ith row vector of matrix A, corresponding to vertex v_i, will be indicated A_i.

Given a weighted graph [G,c], where c is a vector of edge weights, define a matching M in G as a subset of the edges such that no more than one edge of M is incident to any node of G. The maximum matching problem is to find a matching of maximum weight sum. This can be written as an integer program as:

$$\text{Max } c^T x \qquad \text{subject to } Ax \leq 1, \qquad x_i = 0 \text{ or } 1,$$

where x is a vector corresponding to the edges in E, and T denotes a vector transpose. Define a covering C in G as a subset of the edges such that each vertex is incident to at least one edge of the subset. The minimum covering problem is to find a covering of minimum weight sum:

$$\text{Min } c^T x \qquad \text{subject to } Ax \geq 1, \qquad x_i = 0 \text{ or } 1.$$

Considerable work has been devoted to the solution of these closely related problems. Berge [4] proved the connection between a maximum cardinality matching and minimum cardinality covering of G, and proved that a matching is of maximum cardinality if and only if no "augmenting path" exists. Norman and Rabin [11] similarly characterized a minimum cardinality covering, requiring that no "reducing path" exists. Edmonds [6,7] shows how to efficiently search for such "augmenting paths", and his technique will be briefly reviewed in Section 2. This approach was generalized to obtain a maximum weight matching. Several authors, Witzgall and Zahn [16] and Balinski [1], presented a simpler alternative to this search procedure, but these simplifications appear to be effective only for the cardinality matching problem, whereas the general weighted case poses extra difficulties.

Edmonds and Johnson [8] have developed an efficient algorithm to solve the general problem for a weighted graph [G,c]:

$$\text{Min } c^T x \qquad \text{subject to } b_1 \leq Ax \leq b_2,\ x_1 = 0 \text{ or } 1,$$

which obtains a maximum matching or minimum covering as a special case. A good discussion of this algorithm for matching has been provided recently in a text by Garfinkel and Nemhauser [10]. The importance of this algorithm in the field of integer programming has been discussed by Balinski [2].

The maximum k-matching problem differs from the maximum matching problem only in that the constraint $\Sigma x_i = k$ is added. White [12] has proved that this problem can be solved by subtracting λ from each edge weight of G, and then obtaining a maximum matching for this transformed graph $[G,c-\lambda]$. The parameter λ may assume any real value, and can be interpreted as a dual variable or Lagrange multiplier as discussed by Everett [9], corresponding to the k-cardinality constraint. All feasible values of k can be achieved by this process, and the weights of the k-matchings obtained in $[G,c]$ can be shown to be concave with parameter k. A review of these results is given in Section 3. The difficulty with this approach is that given k, λ must be found by some systematic single dimensional search strategy.

The maximum k-matching algorithm described in Section 4 generates a maximum k-matching as k sweeps through all feasible values, and produces an appropriate value of λ in the process. This algorithm obtains an optimum k-matching at every stage of the process, obtaining a maximum matching as a special case. A complexity analysis shows that only marginally more work is required for this result than for the Edmonds-Johnson algorithm, and computational experience is cited in Section 5.

A similar parametric approach has been applied to the minimum covering problem, and minimum k-covers are characterized by White in [13]. There are some aspects which make the minimum k-cover problem more complex than k-matching, and an algorithm is presented in White and Gillenson [14]. An interesting insight into the design of "star networks" in communication networks results from this algorithm and has been presented in White [15].

2. SYSTEMATIC SEARCH FOR AUGMENTING PATHS

In order to obtain augmenting paths, the following definitions are required relative to a given matching M in graph G. If no edge of M is incident to vertex v, then v is called an *exposed vertex* (relative to M). An *alternating path* P is a path whose edges are alternately in M and not in M. In order to form a new matching M' by exchanging edges in a matching M with those edges not in a matching along some alternating path P, the symmetric difference notation, M⊕P, is used:

$$M' = M \oplus P = (M-P) \cup (P-M).$$

An *augmenting path* is an alternating path between two exposed vertices, relative to some matching in a graph.

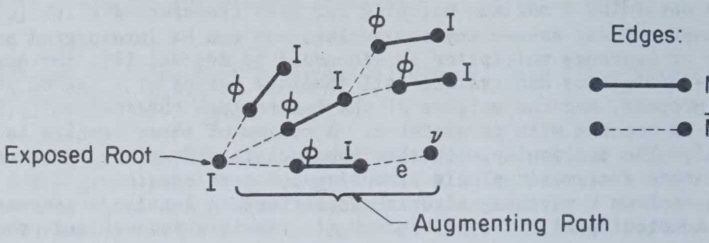
Berge [4] proved a matching M is of maximum cardinality in a graph G if and only if there exists no augmenting path in G. Efficient search structures are required to find such augmenting paths. An *alternating tree* relative to matching M is a tree with the added conditions that:

(1) exactly one vertex, the root of the tree, is exposed;
(2) all paths from the root are alternating paths;
(3) all maximal paths from the root are of even length.

Label the vertices in the alternating tree as *even* I and *odd* Ø, depending upon their distance from the root within the tree; the vertices are so labelled in Figure 1.

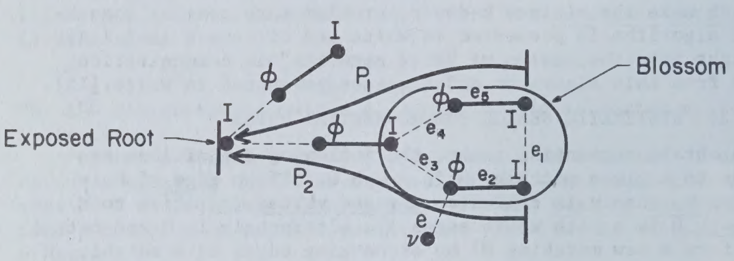
An *augmenting tree* is an alternating tree with one additional edge e from a terminal even vertex to some exposed vertex v not in the tree. An augmenting tree is shown in Figure 1. This edge e uniquely determines an augmenting path from v to the root.

Edmonds [6,7] recognized this simple search strategy must be modified in the presence of odd length circuits in the graph. As shown in Figure 2, depending upon how the tree was formed, an augmenting path might be found from an odd Ø vertex, thus excessively complicating the search. Observe that if the set of vertices comprising this odd circuit, together with all edges with both endpoints in this set, are conceptually shrunk as a "blossom" to a single pseudonode, and this pseudonode labelled as I, the augmenting path can be found as before. By shrinking blossoms in this manner, the concept of an alternating tree can be preserved in the resulting graph.



Edges:
M
M̄
ϕ
I
ϕ
I
ϕ
I
ϕ
I
ϕ
I
ϕ
I
ν
e
Exposed Root
I
Augmenting Path

FIGURE 1 Augmenting Tree



I
ϕ
P1
Blossom
ϕ
e5
I
Exposed Root
I
ϕ
I
e4
e1
e3
ϕ
e2
I
P2
e
ν

FIGURE 2 Forming a Blossom Within an Alternating Tree

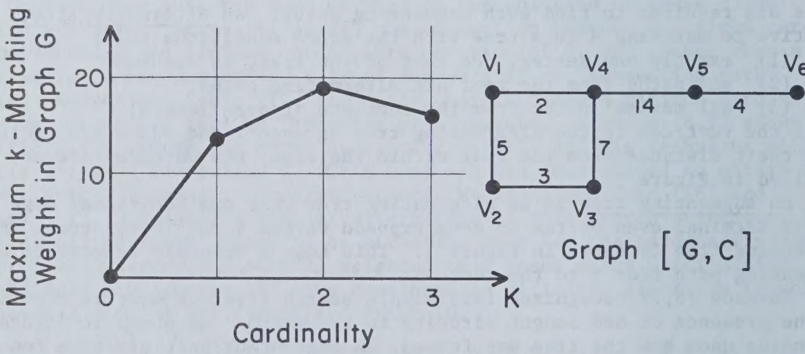


FIGURE 3 Maximum k-Matching Curve for Graph [G,c]

Another way to see the difficulty presented by odd circuits, and the necessity for "blossoms", as provided by Balinski [1], where he showed that all vertices of the convex polyhedron defined by

$$Ax \leq 1,\ x \geq 0,$$

have coordinate values 0, 1/2, or 1. For example, consider the subgraph defined by the circuit in Figure 2 defined by $\{e_1, e_2, e_3, e_4, e_5\}$. For unity weights $c_i = 1$, a maximum weight matching has the value of two, whereas the noninteger solution $x_i = 1/2$ satisfies the constraints $Ax \leq 1$, $x \geq 0$, but with value $c^T x = 5/2$.

Edmonds' identification of "blossoms" is equivalent to the addition of blossom constraints in lieu of the explicit integrality requirement, which will serve to "cut" the noninteger extreme points from the closed convex solutions of $Ax \leq 1$, $x \geq 0$. Let S_p be a set of $(2r_p + 1)$ vertices, and let R_p be a vector such that

$$R_{pi} = \begin{cases} 1, & \text{if edge } e_i \text{ has both endpoints in set } S_p, \\ 0, & \text{otherwise.} \end{cases}$$

Then $R_p x \leq r_p$ is a linear constraint which will "cut" a noninteger extreme point. If R is the odd vertex set-edge incidence matrix, this can be written as

$$Rx \leq r,$$

where r is the appropriate vector of odd vertex set indices. Although there is an extremely large number of constraints, this representation suffices to convert the matching integer program to a linear program, and Edmonds [6,7] proves the formulations equivalent. An alternative proof is provided by Balinski [3]. More importantly, the blossom constraints need only be considered as blossoms are invoked in the matching algorithm, where it can be shown that very few constraints are binding at any time.

These constraints are invoked in the matching algorithm by identifying a blossom B_p as a set S_p of $(2r_p + 1)$ vertices, and a set of edges R_p which have both endpoints in S_p. Relative to a matching M, define a blossom recursively as

a) a blossom always satisfies the corresponding constraint $R_p x \leq r_p$ with equality;

b) a blossom is formed by an alternating odd circuit relative to M and involves vertices in this circuit; vertex set S_p is conceptually shrunk to a pseudonode, and incidences re-established relative to vertices outside S_p;

c) a blossom is formed by an alternating odd circuit relative to M, and involves both vertices or pseudonodes in this circuit; a pseudonode is constructed as in b).

Note that the blossom in Figure 2 satisfies these conditions, specifically that $r_p = 2$, $|S_p| = 5$, and there are two matching edges within the identified odd circuit. For further details concerning blossom constraints, see Edmonds [6,7], Balinski [3], or Garfinkel and Nemhauser [10].

3. A PARAMETRIC APPROACH TO K-MATCHING

In this section, the parametric approach to maximum matching will be developed. This theory will be applied in Section 4 with the description of an efficient maximum k-matching algorithm. The parametric approach consists of transforming a given graph $[G,c]$ to a graph $[G,c-\lambda]$ by subtracting λ from each edge weight of G. If M is a matching in graph G, let W(M) represent the weight of matching M in $[G,c]$ whereas $W_\lambda(M)$ represents this weight in $[G,c-\lambda]$. Let $|M|$ denote the cardinality of set M.

Lemmas 1 and 2 below show how maximum k-matchings are related to graph $[G, c-\lambda]$, and that k values are monotonic with λ.

LEMMA 1 If graph $[G,c-\lambda]$ has a maximum matching M, and $|M| = k$, then M is a maximum k-matching in $[G,c]$.

PROOF Let M' be any k-matching in G; then for any λ,

$$W_\lambda(M) = W(M) - k\lambda,$$

so $W_\lambda(M) \geq W_\lambda(M')$ implies $W(M) \geq W(M')$.

LEMMA 2 Given $\lambda_2 > \lambda_1$, and let M_1 and M_2 be maximum matchings in $[G, c-\lambda_1]$ and $[G, c-\lambda_2]$, respectively. Then if $|M_1| = k_1$, $|M_2| = k_2$, $k_1 \geq k_2$.

PROOF By assumption

$$W_{\lambda_1}(M_1) \geq W_{\lambda_1}(M_2),\ W_{\lambda_2}(M_2) \geq W_{\lambda_2}(M_1).$$

Expanding,

$$W(M_1) - k_1\lambda_1 \geq W(M_2) - k_2\lambda_1,$$
$$W(M_2) - k_2\lambda_2 \geq W(M_1) - k_1\lambda_2.$$

Adding and rearranging, with $\lambda_2 - \lambda_1 \neq 0$, yields $k_1 \geq k_2$.

Lemmas 1 and 2 show the potential of the parametric approach for the solution of the maximum matching problem. The difficulty is that not all values of k may be obtained as continuous values of λ are examined. Everett [9] refers to this problem as "gaps" in the Lagrange multiplier method. Theorem 1 below shows that no such "gaps" occur in the proposed technique. In the sequel let m denote the maximum cardinality of any matching in graph G. If M_1 and M_2 are two matchings, and P is an augmenting path in $M_1 \oplus M_2$ with respect to M_1, then define

$$W(P) = \sum_{e_i \in P\cap M_2} c_i - \sum_{e_i \in P\cap M_1} c_i$$

THEOREM 1 Given $0 \leq k \leq m$, there exists a λ such that some maximum matching in $[G, c-\lambda]$ is of cardinality k.

PROOF Assume that such a "gap" exists, and thus a value λ exists such that matchings M_1 and M_2 are both maximum in $[G, c-\lambda]$, with $|M_1| = k_1$, $|M_2| = k_2$, $k_1<(k_2-1)$, but no k-matching is maximum in $[G,c-\lambda]$ for all k such that $k_1<k<k_2$.

Within edge set $M_1 \oplus M_2$ there exists an augmenting path P relative to M_1; this can be constructed from some vertex exposed relative to M_1 but not with respect to M_2. P extends to another vertex of similar description, and note this must be a vertex distinct path, since M_1 and M_2 are matchings. Form matchings $M_1 \oplus P$ and $M_2 \oplus P$ in G, and note

$$W_\lambda(M_1 \oplus P) = W_\lambda(M_1) + W(P) - \lambda < W_\lambda(M_1),$$
$$W_\lambda(M_2 \oplus P) = W_\lambda(M_2) - W(P) + \lambda < W_\lambda(M_2);$$

but since $W_\lambda(M_1) = W_\lambda(M_2)$, this leads to a contradiction, and proves the theorem.

It can be shown that Theorem 1 can only be valid for the k-matching problem if the weight sums of maximum k-matchings are concave with k, or else this is precisely where "gaps" can occur. Theorem 2 below proves all such curves concave; one implication of concavity is that such functions are unimodal. A typical concave curve for an example graph $[G,c]$ is shown in Figure 3.

THEOREM 2 The weight sums of maximum k-matchings in $[G,c]$ are concave with k.

PROOF Choose any $1 \leq k \leq (m-1)$, and let M_1, M_2, and M_3 denote selected maximum (k+1)-, k-, and (k-1)-matchings respectively in [G,c]. Theorem 1 assures us that all such k-matchings exist. If M_2 is maximum in graph [G, c-λ], then

$$W_\lambda(M_2) \geq W_\lambda(M_1), \quad W_\lambda(M_2) \geq W_\lambda(M_3).$$

Expanding,

$$W(M_2) - k\lambda \geq W(M_1) - (k-1)\lambda,$$

$$W(M_2) - k\lambda \geq W(M_3) - (k+1)\lambda,$$

and adding these two inequalities, yields,

$$W(M_2) - W(M_3) \geq W(M_1) - W(M_2),$$

which proves the weights of k-matchings are concave with k in [G,c].

In the k-matching algorithm presented in Section 4, maximum k-matchings are obtained for sequential values of k via a single augmenting path, without the necessity of obtaining intermediate matchings. Theorem 3 shows how this is possible.

THEOREM 3 For $0 \leq k \leq (m-1)$, let M_1 be a maximum k-matching in [G,c], and M_2 a maximum (k+1)-matching in [G,c] such that $M_1 \oplus M_2$ is minimal. Then subgraph $M_1 \oplus M_2$ consists of exactly one augmenting path with respect to M_1.

PROOF Consider an augmenting path P with respect to M_1, which most certainly must exist in $M_1 \oplus M$. Consider matchings $M_2 \oplus P$ and $M_1 \oplus P$, which are of cardinalities k and (k+1) respectively, so

$$W(M_2 \oplus P) \leq W(M_1), \quad W(M_1 \oplus P) \leq W(M_2).$$

Expanding,

$$W(M_2) - W(P) \leq W(M_1),$$

$$W(M_1) + W(P) \leq W(M_2),$$

which implies $W(P) = W(M_1) - W(M_2)$, and thus the minimality of $M_1 \oplus M_2$ requires no edges be included in $M_1 \oplus M_2$ other than augmenting path P.

4. THE K-MATCHING ALGORITHM

Given a weighted graph [G,c], the maximum k-matching problem is to find a maximum matching M in G with cardinality $|M| = k$, and the algorithm described in this section will achieve these configurations as k monotonically moves through all feasible values. The motivation for the algorithm is based on the following primal-dual linear programming system:

Primal Max $c^T x$ subject to

$$Ax \leq 1, \qquad Rx \leq r,$$

$$\Sigma x_i = k, \qquad x \geq 0$$

where the need for the blossom constraint $Rx \leq r$ was discussed in Section 2.

Dual Min $[\Sigma\, y_i + \lambda k + \Sigma r_p z_p]$ subject to

$$y_i + y_j + \lambda + \sum_{e_{ij} \varepsilon R_p} z_p \geq c_{ij} \qquad \text{for all } c_{ij} \varepsilon E$$

$$y \geq 0, \; z \geq 0, \; \lambda \text{ unconstrained.}$$

In these dual constraints, λ is the same Lagrange multiplier corresponding to the primal constraint $\Sigma x_i = k$ as discussed in Section 3. Note that a maximum matching M in [G,c] is obtained when $\lambda = 0$. Then $\lambda \geq 0$ will obtain matchings of cardinality $k<|M|$, and $\lambda \leq 0$ yields matchings of cardinality $k>|M|$. The dual variable y_i corresponds to a vertex $v_i \varepsilon V$, but in the algorithm it will be convenient to define a modified vertex weight

$w_i = y_i + \lambda/_2$.

It can be proven that the k-matching algorithm obtains optimal k-matchings at each stage because both primal-dual feasibility and orthogonality (complementary slackness) are maintained at each step. This is shown in reference [12], and the orthogonality conditions in this case are:

$x_{ij} > 0 \Rightarrow w_i + w_j + \sum_{e_{ij} \varepsilon R_p} z_p = c_{ij}$,

$w_i > \lambda/_2 \Rightarrow A_i x = 1$,

$z_p > 0 \Rightarrow R_p x = r_p$.

The algorithm proceeds by growing a forest of alternating trees, one tree rooted at each vertex exposed relative to the current matching. By growing trees in parallel, the augmenting path of largest possible weight is obtained - except for ties, this is the augmenting path identified in Theorem 3 of Section 3. Each exposed node serving as the root for such a tree has weight $w_i = \lambda/_2$ at each stage of the algorithm. As described in Section 2, a vertex or blossom in a tree is labelled even (I) or odd (∅) depending upon its distance from the root within the tree. A neutral (NU) vertex or blossom is incident to an edge currently in the matching, but not presently in any tree. All vertices within a blossom carry the same label as that blossom. For convenience, let I, ∅, and NU denote a partition of the vertices V according to their labels.

An overview of the k-matching algorithm is shown in Figure 4. A summary of the steps of this algorithm are as follows, where the details can be found in Reference [12]:

1) Begin the algorithm with an empty matching, k = 0, all $z_p = 0$, since no blossoms B_p have yet formed, and

   $\lambda = \operatorname{Max}_{e_{ij} \varepsilon E} [c_{ij}]$, $w_i = \lambda/_2$ for each vertex $v_i \varepsilon V$.

2) Grow a forest of alternating trees, each tree rooted at an exposed vertex, and classify all such vertices as I.

3) For all edges $e_{ij} \varepsilon E$, calculate Δ as

   $\Delta = \operatorname{Min} [\Delta_1, \Delta_2, \Delta_3, \Delta_4]$ where each Δ_i is called valid when

   $\Delta_1 = 1/2 \operatorname{Min}_{B_p} [z_p]$, where blossom B_p is labelled ∅, but is outermost in a nesting of blossoms.

   $\Delta_2 = 1/2 \operatorname{Min}_{e_{ij} \varepsilon E} [c_{ij} - w_i - w_j]$, where v_i, v_j ε I, but are in different trees.

   $\Delta_3 = \operatorname{Min}_{e_{ij} \varepsilon E} [c_{ij} - w_i - w_j]$, where v_i ε I, $v_j \varepsilon NU$.

   $\Delta_4 = 1/2 \operatorname{Min}_{e_{ij} \varepsilon E} [c_{ij} - w_i - w_j]$, where v_i, v_j ε I, but are in the same tree yet do not belong to the same blossom B_p.

4) If no valid Δ exists, then no (k+1)-matching exists in graph G, and the algorithm is terminated.

5) Upon calculation of a valid Δ, adjust the dual variables as follows

   $w_i \leftarrow w_i - \Delta$, for $v_i \varepsilon I$

   $w_i \leftarrow w_i + \Delta$, for $v_i \varepsilon \emptyset$

   $z_p \leftarrow z_p + 2\Delta$, only for outer most blossoms B_p which are labelled I

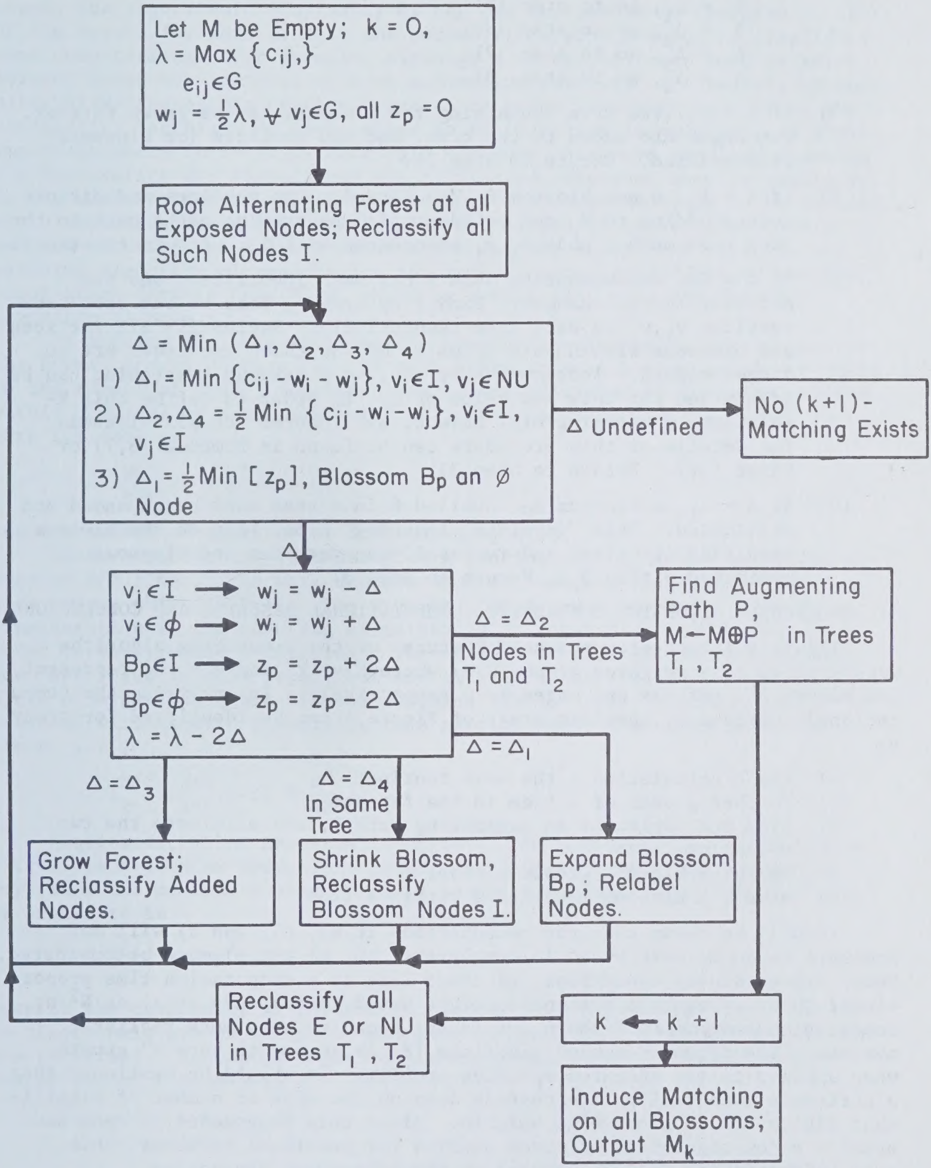

Let M be Empty; k = 0,
λ = Max {cij,}
eij∈G
wj = ½λ, ∀ vj∈G, all zp = 0
Root Alternating Forest at all Exposed Nodes; Reclassify all Such Nodes I.
Δ = Min (Δ1, Δ2, Δ3, Δ4)
1) Δ1 = Min {cij - wi - wj}, vi∈I, vj∈NU
2) Δ2, Δ4 = ½ Min {cij - wi - wj}, vi∈I, vj∈I
3) Δ1 = ½ Min [zp], Blossom Bp an ∅ Node
Δ Undefined
No (k+1)-Matching Exists
Δ
vj∈I → wj = wj - Δ
vj∈∅ → wj = wj + Δ
Bp∈I → zp = zp + 2Δ
Bp∈∅ → zp = zp - 2Δ
λ = λ - 2Δ
Δ = Δ2
Nodes in Trees T1 and T2
Find Augmenting Path P, M←M⊕P in Trees T1, T2
Δ = Δ1
Δ = Δ3
Δ = Δ4
In Same Tree
Grow Forest; Reclassify Added Nodes.
Shrink Blossom, Reclassify Blossom Nodes I.
Expand Blossom Bp; Relabel Nodes.
Reclassify all Nodes E or NU in Trees T1, T2
k ← k + 1
Induce Matching on all Blossoms; Output Mk

FIGURE 4 Algorithm for Maximum k-Matching M_k in Graph [G,c].

$z_p \leftarrow z_p - 2\Delta$, only for outermost blossoms B_p which are labelled 0

6) Then if
$\Delta = \Delta_1$, go to step 10);
$\Delta = \Delta_2$, go to step 9);
$\Delta = \Delta_3$, go to step 7);
$\Delta = \Delta_4$, go to step 8).

7) If $\Delta = \Delta_3$, the tree containing the I vertex can be grown further. Two edges are added to the tree, and two vertices (or blossoms) reclassified. Return to step 3).

8) If $\Delta = \Delta_4$, a new blossom B_p has formed. The vertices and circuit corresponding to B_p can be identified by tracing paths back to the root from each I node v_i, v_j associated with Δ_4. Return to step 3).

9) If $\Delta = \Delta_2$, an augmenting path P has been identified, and the matching can be changed. Find P by tracing back to the root from vertices v_i, v_j in each tree respectively. Reclassify all vertices and blossoms within both trees as NU, as these two trees are no longer needed. Increment k by one, as a maximum k-matching can be identified for this new value of k. In order to define this k-matching, a matching will have to be "induced" on all blossoms. The details of this procedure can be found in Edmonds [6,7] or White [12]. Return to Step 3).

10) If $\Delta = \Delta_1$, a blossom B_p labelled 0 in a tree must be expanded and eliminated. This involves "inducing" a matching on the blossom, repairing the tree, and reclassifying vertices and blossoms contained within B_p. Return to Step 3).

5. K-MATCHING ALGORITHM COMPLEXITY, COMPUTATIONAL RESULTS, AND CONCLUSIONS.

Figure 4 illustrates the main features of the k-matching algorithm when applied to a weighted graph [G,c], where $N = |V|$ and $n = |E|$ represent the number of vertices and edges in G respectively. In analyzing the computational complexity, specific areas of Figure 4 can be identified for study as

a) the Δ calculation - the main routine;
b) further growth of a tree in the forest;
c) find and implement an augmenting path P, and eliminate the two trees involved;
d) shrink a blossom within a tree;
e) expand a blossom labelled 0 within a tree.

It will be shown that the computations in b), c), and d) will not dominate an asymptotic bound in complexity, but a) and e) must be considered. Under extraordinary conditions, e) could lead to a computation time proportional to N^4, whereas a reasonably tight bound corresponds to a) as $N^2 \cdot n$; computational experience shows the latter bound to be a more realistic measure. The Edmonds-Johnson algorithm [8] is argued to have N^4 growth when applied to the weighted matching problem. It should be mentioned that algorithmic computation time depends also on the size or number of significant digits of the given edge weights. Since this dependence is weak and usually a few significant figures suffice for practical problems, this edge weight factor will be ignored in the subsequent discussion.

The complexity bounds depend heavily on how a matching algorithm is coded. The k-matching algorithm has been carefully coded for efficiency, with data structures designed especially to reduce the computations in routines b), c), and d). Throughout this discussion it should be kept in mind that the k-matching algorithm obtains a maximum k-matching for all feasible values of k, and the best possible augmenting path is found at every stage, whereas implementations of the Edmonds-Johnson algorithm obtain a meaningful configuration only when a final maximum matching is achieved. Although a forest of alternating trees must be grown in parallel at every stage, the algorithmic complexity of the two approaches is comparable. Part of the reason for this is that the parametric approach actually simplifies some computations. For example, although a forest of trees must be maintained, trees not involved with an augmenting path can be retained, whereas alternative approaches require trees to be grown from scratch. Also, the cardinality increases at every stage as the matching is modified. A most important characteristic is that only blossoms which are absolutely needed for integrality are formed, and are eliminated only when that is necessary. In the Edmonds-Johnson algorithm, blossoms may be formed only to be immediately expanded, since they were really not needed in order to achieve a maximum matching. A related observation is that except for ties, the k-matching algorithm always obtains the same results, regardless of how the input graph data is specified, which is a desirable property for an algorithm to have.

In arguing the complexity of the k-matching algorithm, first consider the forest growth routine b). At each stage at most k edge pairs are added to the forest of trees, requiring the reclassification of 2k nodes, whether within blossoms or not. Since $k \leq N/2$, an upper bound on the amount of work for routine b) is

$$\Sigma_{k=1}^{N/2} 2k = N^2/4 + \frac{N}{2},$$

and the number of calls of b) is bounded by $N^2/8$.

Next, consider the augmenting path routine c), which may be called as many as N/2 times. The work depends upon the maximum number of nodes in both trees, which is 2k+2. The work here then grows as $N^2/4$. Another consideration is the inducing of matchings on edges within blossoms, in order to specify the k-matching. The details of this procedure have not been specified in this paper, but are in References [6], [8], [10], or [12]. Since a largest possible blossom contains (2k+1) nodes, a nested set of k blossoms will require traversing this partial order a maximum number of times, i.e.,

$$\Sigma_{k=1}^{N/2} \Sigma_{i=0}^{k-1}(k-i) \approxeq \frac{N^3}{48}.$$

Routine d), which involves identifying and shrinking a blossom within a tree, is a complex operation. Since there are at most (2k+1) nodes in a any tree, the maximum number of calls to this routine over all stages can be argued to be

$$\Sigma_{k=1}^{N/2} k = N^2/8.$$

Despite the complexity of blossom routine d), the significant contribution to algorithmic growth is due to updating all nodes as the blossom elements are traversed. It can be shown that this is maximized for a maximally nested set of k blossoms, or

$$\Sigma_{k=1}^{N/2} \Sigma_{i=0}^{k-1} N \approx \frac{3N^3}{8}.$$

Routine e) poses another complex operation as a blossom labelled 0 within a tree must be expanded. If we assume that all blossoms formed must be expanded, the maximum number of calls to this routine is the same as for d). The most significant contribution to complexity is that every node within the expanded structure must be updated and the partial ordering of blossoms used to discover the new outermost blossom. This work can be shown to be maximized when the nesting is maximal at each stage, so at the i-th expansion, there are (2k-2i+1) nodes involved, with the partial ordering nesting level at (k-i). The work in this extraordinary case is

$$\Sigma_{k=1}^{N/2} \Sigma_{i=0}^{k-1}(2k-2i+1)\cdot(k-i) \cong \frac{N^4}{24} .$$

Notice that this growth was obtained only when all blossoms formed are maximally nested within a single tree, and then must be totally expanded, and this procedure repeated at every stage.

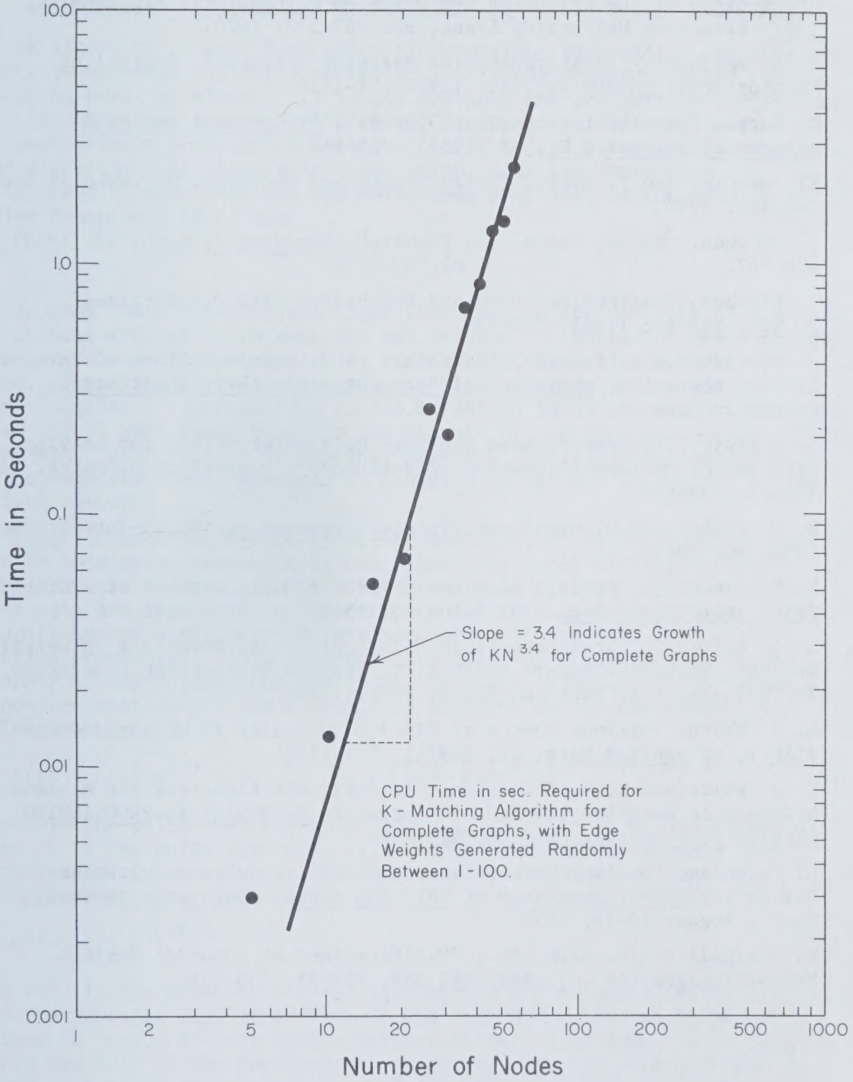
Figure 4 shows that the number of times the main routine a) is executed is the sum of the calls on b), c), d), and e). This has already been argued to be on the order of $3N^2/8$. The work within routine a) depends upon the calculation of Δ. Each I node must be scanned and all edges examined which emanate from that node. This work is bounded by n and accounts for the work in calculating Δ_2, Δ_3, and Δ_4. The total complexity of a) is then $3N^2 \cdot n/8$.

In summary, the computation time of the k-matching could grow as N^4, but this corresponds to an unrealistic situation. On the other hand, the next alternative growth is $N^2 \cdot n$, which is quite realistic for many graph configurations. For example, even if no blossoms are formed or expanded, ordinary tree growth could give rise to the achievement of this bound. Storage requirements grow linearly with n.

This algorithm has been coded in FORTRAN and used with an IBM 370/165. Figure 5 shows the results of experiments with complete graphs for up to 55 vertices, with edge weights generated uniformly from 1-100. The figure shows algorithmic growth of $N^{3.4}$, which is reasonably close to the suggested bound. The data below show a comparison with the Edmonds-Johnson algorithm for randomly generated graphs; this data is well within the $N^2 \cdot n$ bound.

N	n	E-J, IBM 360/91 CPU Time, Sec	k-Match, IBM 370/165 CPU Time, Sec
60	300	9	0.53
100	500	27	1.70
140	700	35	3.96
200	1000	73	7.65
300	1500		18.83

In conclusion, this paper has considered the problem of obtaining maximum k-matchings for a given weighted graph [G,c]. The parametric approach to this problem has been described, and it was shown that no "gaps" occur in the use of a Lagrange multiplier for this problem. The weight sums of the maximum k-matchings in [G,c] have been shown to be concave with k. Based on the techniques of Edmonds, an efficient algorithm to implement this parametric approach has been described, requiring computation time on the order of $N^2 \cdot n$. This algorithm efficiently implements the parametric approach, obtaining a maximum k-matching and appropriate Lagrangian multiplier λ for all feasible values of k within this computational bound, and obtains a maximum matching in [G,c] as a special case.



Slope = 3.4 Indicates Growth of $KN^{3.4}$ for Complete Graphs

CPU Time in sec. Required for K-Matching Algorithm for Complete Graphs, with Edge Weights Generated Randomly Between 1-100.

FIGURE 5 Computer Time Required for k-Matching Algorithm on Complete Graphs.

REFERENCES

1. M. L. Balinski, "Labelling to Obtain a Maximum Matching", Proceedings of the Conference on Combinatorial Mathematics and Its Applications, pp. 585-602, University of North Carolina, April, 1967.
2. M. L. Balinski, "On Recent Developments in Integer Programming", Proceedings of the Princeton Symposium on Mathematical Programming, 1967, Princeton University Press, pp. 267-302, 1970.
3. M. L. Balinski, "Establishing the Matching Polytope", Journal of Combinatorial Theory (B), 13, (1972), 1-13.
4. C. Berge, "Two Theorems in Graph Theory", Proceedings National Academy of Science U.S., 43 (1957), 842-844.
5. R. Busacker and T. Saaty, Finite Graphs and Networks, McGraw-Hill Book Company, 1965.
6. J. Edmonds, "Paths, Trees, and Flowers", Canadian J. Math, 17 (1965), 449-467.
7. J. Edmonds, "Maximum Matching and Polyhedron with 0,1-Vertices", J. Res. NBS 69B (1965) 125-130.
8. J. Edmonds and E. Johnson, "Matching: A Well-Solved Class of Integer Linear Programs", Combinatorial Structures and Their Applications, Gordon and Breach, 1970, pp. 89-92.
9. H. Everett III, "Generalized Lagrange Multiplier Method for Solving Problems of Optimum Allocation of Resources", Operations Research, 11, (1963), 399-417.
10. R. Garfinkel and G. Nemhauser, Integer Programming, Wiley-Interscience, 1972, pp. 88-101.
11. R. Norman and M. Rabin, "An Algorithm for a Minimum Cover of a Graph", Proc. Amer. Math. Soc., 10, (1959) 315-319.
12. L. J. White, "A Parametric Study of Matchings and Coverings in Weighted Graphs", Technical Report 06920-11-T, Systems Engineering Laboratory, The University of Michigan (Ph.D. Thesis) 1967.
13. L. J. White, "Minimum Covers of Fixed Cardinality in Weighted Graphs", SIAM J. of Applied Math, 21, (1971), 104-113.
14. L. J. White and M. L. Gillenson, "An Efficient Algorithm for Minimum k-Covers in Weighted Graphs", to appear in MATHEMATICAL PROGRAMMING STUDIES #2: Integer Programming, 1974.
15. L. J. White, "Optimum Concentrator Location in Telecommunications Design", SIGMAP Proceedings of 1972 ACM Annual Conference, Boston, Mass., August 14-16, 1972.
16. C. Witzgall and C. Zahn, Jr., "Modifications of Edmonds' Maximum Matching Algorithm", J. Res. NBS 69B, (1965), 130-135.

A NONLINEAR INTEGER PROGRAMMING ALGORITHM FOR THE DESIGN OF RADAR WAVEFORMS

ARTHUR I. COHEN
Systems Control, Inc.
Palo Alto, California

ABSTRACT

An algorithm is described which finds optimum transmitter and receiver weights to maximize clutter suppression in a predetermined clutter region when using burst waveforms. It is assumed that the transmitter weights can only take on values from a finite set. This optimization problem is a nonlinear integer programming problem. It is solved using a branch and bound algorithm. An example is given which shows the improvement in clutter suppression when this new design procedure is used as compared to a simpler nonoptimal technique.

I. INTRODUCTION

In many radar applications there will lie, in the vicinity of the target, objects from which returns are not desired. Examples of these objects, which are called clutter, are rain, rain clouds, dust clouds, or the ground. The standard radar detection problem is to determine whether a return contains a component due to the target or is only due to clutter and noise. It is well known that, in the absence of clutter, the matched filter is optimum. When clutter is present, however, the transmitted waveform and the receiver can be designed to minimize the effect of clutter on the receiver output.

In order to abrogate the effect of the clutter, it is necessary that there be some prior knowledge of characteristics that distinguish the target from the clutter. In this paper we shall assume that the clutter and targets are separated in velocity. Under this condition the N-pulse uniformly spaced burst waveform has been shown [1] to have good clutter rejection properties. This waveform is a train of N identically shaped uniformly spaced pulses; the amplitude and phase relation of the pulses are however arbitrary. The transmitted waveform can be represented in complex form by

$$s(t) = u(t)\, e^{j2\pi f_o t} = \sum_{n=1}^{N} x_n\, r\left(\frac{t-nT}{\Delta}\right) e^{j2\pi f_o t} \tag{1}$$

where T is the pulse spacing, Δ is the pulse duration, N is the number of pulses, r is the pulse waveform, $\{x_n\}$ are the transmitter weights and f_o is the transmitted frequency. If a matched filter receiver were used, its impulse response would be

$$s^*(-t-t')\, e^{j2\pi f_1 t} \tag{2}$$

where $t=t'$ is the expected arrival time of the signal and f_1 differs from f_o by the expected doppler shift. In the case where clutter is present, and thus it is not evident that a matched filter is optimum, it is natural to keep the form of the receiver the same; that is, the receiver will be assumed to have an impulse response $g(t) = h^*(-t-t')$ where $h(t)$, the weighting function of the receiver, is of the form

$$h(t) = v(t)\, e^{j2\pi f_1 t} = \sum_{n=1}^{N} w_n\, r\left(\frac{t-nT}{\Delta}\right) e^{j2\pi f_1 t} \tag{3}$$

where w_n are the receiver weights and f_1 differs from the transmitter frequency f_o by the doppler frequency to which the receiver is tuned. The output of the receiver in response to a signal $y(t)$ from a target moving at a velocity corresponding to a doppler shift f_s at a range corresponding to a time delay t' is then

$$|z(t)| = \left| \int u(\eta) v^{*} (\tau + \eta) e^{-j2\pi f \eta} d\eta \right| \tag{4}$$

where

$\tau = t - t'$

$f = f_o - f_1 - f_s$

The waveform-receiver parameters to be chosen to minimize the effect of clutter are therefore the transmitter weights $\{x_n\}$, the receiver weights $\{w_n\}$, the pulse spacing T and the number of pulses N. Since a uniformly spaced burst has maximum clutter rejection at doppler $\nu = ½T$ the pulse spacing will be chosen so that the clutter lies midway between dopplers $\nu = 0$ and $\nu = 1/T$. The transmitter and receiver weights can then be chosen to either maximize the signal-to-interference ratio (interference = clutter + noise) or minimize the average sidelobe level in the clutter region. N can then be chosen by finding the optimum transmitter and receiver weights for various values of N and then optimizing over N.

Several papers have appeared in recent years concerning weighting, on receive and transmit, of the burst waveform. Rummler [2] developed a method to calculate these weights to optimize the signal-to-interference ratio where a particular clutter distribution was assumed. Delong [3] generalized this procedure to optimize the weights given that they may have random implementation errors. Both of these techniques require that both the transmit and receiver weights can take on any desired value. (The weights can, however, be normalized so that the maximum weight is achievable). Many radars cannot produce any set of weights. For example, some radars can weight on transmit only by turning off one or more of their power tubes, and some phased array radars weight by turning off elements or subarrays. A straightforward design technique for these radars is to take the weights designed using the aforementioned methods and then quantize the transmitter weights to their nearest value. It shall be shown, however, that this technique yields weights that are far from optimal.

In this report, a generalization of the Rummler design procedure is presented, which finds an optimum set of weights (to maximize the signal-to-interference ratio or average sidelobe level), given an assumed clutter region where the transmitter weights must be real and belong to a given finit set. The clutter region is assumed to be situated midway between dopplers 0 and 1/T. The design problem, which is a nonlinear integer programming problem, is solved using a branch and bound method. The time it takes the method to reach an optimal solution depends on the number of pulses in the burst and number of allowed values for the weights. For a 15 pulse burst with 25 allowable weights, the method typically converged to an optimal solution in less than a minute on the UNIVAC 1108.

The design problem has the option of requiring that the transmitter and receiver weights be symmetric (i.e., if x_i, w_i $i = 1, 2, \ldots, N$ are respectively the transmitter and receiver weights, then symmetric weights require that $x_i = x_{N+1-i}$ and $w_i = w_{N+1-i}$). One advantage of this requirement is that at most N/2, if N is even, or (N+1)/2, if N is odd, transmitter and receiver weights have to be implemented. Another advantage is that the optimization algorithm converges to a solution in a much shorter time than for the full problem, since there are approximately half as many unknowns. An example shall be given showing the effect of assuming symmetry. (The mathematical development for the symmetric case is given in [4]).

The paper is organized as follows: Section II defines the optimization criteria average sidelobe level, as a function of the receiver and transmitter weights and the assumed clutter region. Section III describes the branch and bound algorithm, and Section IV presents some numerical results for two assumed quantizations.

II. OPTIMIZATION CRITERIA

The criteria considered in this paper is the average sidelobe level in the clutter region. It is shown in [4] that the methods developed in this paper can also be used when the sighal-to-interference ratio is used as the criteria.

Average Sidelobe Level

Define the normalized r.m.s. value of the k^{th} sidelobe of the ambiguity function at a particular doppler ν, to be

$$S(k,\nu) = \frac{\int_{(k-\frac{1}{2})T}^{(k+\frac{1}{2})T} |\Psi(\nu,\tau)|^2 d\tau}{\int |\chi(\nu,\tau)|^2 d\tau} \tag{5}$$

where Ψ is the cross ambiguity function of the burst waveform-receiver (1), (3) and χ is the ambiguity function of the pulse r. This is the clutter rejection obtained at the k^{th} sidelobe by using a burst on an extended target at doppler ν as compared to the rejection from a single pulse r. (The target is assumed to extend over the region $[(k-\frac{1}{2})T, (k+\frac{1}{2})T]$. The average sidelobe level over the whole clutter region (see Fig. 1 for a model of the clutter region) is then defined as

$$S = \frac{1}{(k_2-k_1+1)(\nu_2-\nu_1)} \sum_{k=k_1}^{k_2} \int_{\nu_1}^{\nu_2} S(k,\nu)\, d\nu \tag{6}$$

where $\nu_1 = \frac{1}{2T}(1-\delta)$ and $\nu_2 = \frac{1}{2T}(1+\delta)$. This can be shown to equal

$$S = \frac{<w, D(x)\ w>}{(<w,x>)^2} \tag{7}$$

where D is the N×N matrix defined by

$$D_{ij}(x) = (-1)^{ij} \frac{\sin (i-j)\pi\delta}{(i-j)\pi\delta} \sum_{k=k_1}^{k_2} x_{i-k}\, x_{j-k} \tag{8}$$

and where, by convention, $x_i = 0$ for $i < 1$ and $i > N$.

Note that if $k_1 = -(N-1)$ and $k_2 = (N-1)$ then $(2N-1)S$ is the average clutter rejection, over the doppler frequencies of interest, obtained by using a burst on an extended target where the target extends over all ranges. The average sidelobe level can be rewritten with w replaced by $\hat{x}$ and x by $\hat{w}$, that is

$$S = \frac{<\hat{x}, D(\hat{w})\ \hat{x}>}{(<\hat{x},\hat{w}>)^2} \tag{9}$$

where $\hat{w}_i = w_{N+1-i}$ and $\hat{x}_i = x_{N+1-i}$. This follows from the identity (see [1])

$$\Psi_{uv}(\nu,\tau) = \Psi_{\hat{v}\hat{u}}(\nu,\tau) \tag{10}$$

where Ψ_{uv} is the cross ambiguity function obtained with u as the transmitted waveform and v as the receiver weighting function, and where $\hat{v}$ is v reversed in time (i.e., $v(t) = \hat{v}(NT - t)$).

III. OPTIMIZATION ALGORITHM

In this section, the following problems will be considered:

(P1) Find the optimal receiver weights w corresponding to the transmitter weights x to maximize

$$f(w) = (\langle w,x\rangle)^2/\langle w, D(x)w\rangle \tag{11}$$

(P2) Find the optimal transmitter weights x corresponding to the receiver weights w to maximize

$$\hat{f}(\hat{x}) = (\langle \hat{x},\hat{w}\rangle)^2/\langle \hat{x}, D(\hat{w})\hat{x}\rangle \tag{12}$$

where the elements of x must belong to some finite set, i.e., $x_i \varepsilon A = \{a_1,\ldots,a_M\}$ for all $i=1,2,\ldots,N$.

The basic iteration used to obtain the optimum* weights is shown below.

receiver weights
⬇ solve (P2)
transmitter weights (quantized)
⬇ solve (P1)
receiver weights

This iteration is repeated to obtain the optimum weights.

Solution of (P1)

(P1) is the problem addressed by Rummler [2]. The optimum receiver weights w corresponding to the transmitter weights x are

$$w_{opt} = D^{-1}(x)\, x \tag{13}$$

Solution of (P2)

(P2) is an integer nonlinear programming problem. An exhaustive search of all the possible solutions to (P2) would require M^N evaluations of $\hat{f}$. Since typical values of M and N are greater than 16, this would be prohibitive. A branch and bound method was used to solve (P2). Basically branch and bound is a method to search a tree where each vertex (node) in the tree corresponds to the solution of a problem which is an upper bound to the solution of all problems corresponding to vertices lower on the tree (i.e., all vertices that would be separated from the tree by the vertex's removal). The end vertices (leaves) correspond to feasible solutions--values of $\hat{f}$ at feasible values of w--and therefore, these vertices correspond to lower bounds on the solution (see Figure 2).

---

*The word optimum is used here in a very loose manner in that we will not prove that the transmitter-receiver weights obtained will be global maximizers of f. However, each iteration will produce an improved solution and since, by physical reasoning, f is bounded from above, the values of f will converge to some, though not necessarily optimal, finite value.

The fundamental idea on which branch and bound is based is that if, at any time, the upper bound value at a vertex is lower than any lower bound obtained up to that time, then the feasible solutions corresponding to leaves of the subtree defined by that vertex (the part of the three which is below that vertex) are not optimal and therefore do not have to be searched. A vertex at which this happens is said to be fathomed. (See [5] for further information on the branch and bound algorithm and its application to integer nonlinear programming.)

For solving (P2) each vertex corresponds to the following subproblem

(SP(k,α)) maximize $\hat{f}(\hat{x})$

$$\text{subject to } \hat{x}_i = \alpha_i \qquad i = 1, 2, \ldots, k \tag{14}$$

where $\alpha = (\alpha_1, \ldots, \alpha_k)$ and $\alpha_i \varepsilon A \quad i = 1, \ldots, k$. The other components of $\hat{x}$ are unconstrained. Let $z(k,\alpha)$ denote the value of $\hat{f}$ at the solution of SP(k,α), then

$$z(k',\alpha') > z(k'',\alpha'') \tag{15}$$

for $k' < k''$ and $\alpha''_i = \alpha'_i \quad i = 1, 2, \ldots, k'$. This follows from the fact that

$$\max_{y \varepsilon \Omega'} f(y) \geq \max_{y \varepsilon \Omega''} f(y) \tag{16}$$

if $\Omega' \supset \Omega''$. Also note that the solution to SP(N,α) is just $\hat{f}(\alpha)$ which is a lower bound corresponding to a feasible solution.

The tree is then defined as follows: Given a vertex corresponding to the problem SP(k,α), the set of vertices directly below this vertex correspond to the problems $SP(k+1,\alpha^1)$, $SP(k+1,\alpha^2)$, ..., $SP(k+1,\alpha^M)$, where $\alpha^1 = (\alpha,a_i)$ (see Figure 3). Let the top vertex in the tree correspond to problem (P1), then this branching rule defines the whole tree.

Now that the tree structure has been defined, all that remains is to determine the solution to SP(k,α) and to describe an efficient tree searching strategy (an efficient strategy is one that keeps the number of nodes interrogated or equivalently the number of subproblems solved, to a minimum). The solution to SP(k,α) was derived by Spafford in [6] and is restated in [4]. The tree search strategy is given next.

Tree Search Strategy

As stated earlier, the advantage of the branch and bound method is its ability to eliminate from consideration certain branches of the solution tree. It does this by determining that all the nodes on these branches have solution values less than a known feasible solution value. For quick convergence it is necessary to obtain a branching rule which quickly obtains good feasible solutions. This requires (1) getting to the bottom of the tree as quickly as possible and (2) branching (choosing new vertices to solve) effectively so as to investigate the better feasible solutions early.

The tree search strategy that was used is flow charted in Fig. 4. The top vertex corresponds to the unconstrained problem which we denote as SP(0). An initial feasible solution is obtained by quantizing the solution of the unconstrained problem. This gives an initial lower bound on the solution to (P2). Setting k=1, we then separate this vertex; that is, we solve the subproblems $SP(k,a_1)$, $SP(k,a_2)$, ..., $SP(k,a_M)$ (k corresponds to the level of the successors). If k=N, i.e. the successor solutions correspond to feasible solutions, we update the lower bound and check for live neighbors. If k≠N we check for live successors. If any of the successors are live, we select the successor which has the "best" chance of leading to a good feasible solution as the next vertex of interest. The rule as to which successor to use, called the branch rule, is to pick the node with the highest branch value. The branch value of node j is defined:

$$\mu^j = \begin{cases} -\infty & ; \text{ if vertex fathomed} \\ \text{value of corresponding SP solution value } z^j & ; \text{ if solution } x^j \text{ has components lying between } a_1 \text{ and } a_M \\ -\max(|\overline{\mu}^j - a_M|, |\underline{\mu}^j - a_1|) & ; \text{ otherwise} \end{cases} \qquad (17)$$

where

$$\overline{\mu}^j = \max\,\{a_M, x_1^j, x_2^j, \ldots, x_M^j\}$$

$$\underline{\mu}^j = \min\,\{a_1, x_1^j, x_2^j, \ldots, x_M^j\}$$

One will therefore branch first on the vertex whose corresponding solution has components in the feasible range (i.e. in (a_1, a_M)) and whose solution value is largest. If the components of the solutions fall outside the feasible range then one will choose the vertex whose values lie closest to the feasible range. If there are no live successors (or if the successors are at the last level), then we check the previous level for live nodes. The program terminates when level 0 is reached. The optimum is the solution corresponding to the greatest lower bound.

IV. NUMERICAL RESULTS

This section presents optimum quantized weights for a specific case. In order to show the advantage of using the branch and bound method, these weights are compared with the clutter rejection performance that would be obtained if one simply took the optimum weights (non-quantized) produced by Rummler's technique and quantized them to the nearest allowable value.

Optimum Weight Results

The average sidelobe level was minimized for a specific waveform (15 pulses), clutter region (doppler extent corresponding to $\delta = 1/2$ and range extent corresponding to $k_1 = -14$ and $k_2 = 14$ in Fig. 1) and quantization (allowed values are 1, 2, ..., 25).

Three different sets of quantized weights will be compared. To obtain all three, we started with an optimum set of receiver weights (no quantization) obtained using Rummler's method. The methods to obtain the quantized transmitter weights are

(OWQ) - Optimum Weights Quantized. Obtain optimum transmitter weights using Rummler's method, then quantize to nearest allowable value.

(OQW) - Optimum Quantized Weights. Obtain quantized optimum transmitter weights using branch and bound algorithm.

(OQWS) - Optimum Quantized Weights Symmetric. Obtain symmetric quantized optimum transmitter weights using branch and bound algorithm.

Finally, the optimum receiver weights were found, using Rummler's method, corresponding to the quantized transmitter weights. This corresponds to one step of the basic iteration described in Sec. III. Further steps produced negligible improvement for this example. Note that the program did not always run to completion in the time allotted (1-1/2 minutes on the UNIVAC 1108). In this case, the "optimum" weights presented are the best weights obtained at the termination of the program. We shall call these weights suboptimal.

The transmitter weights for the three quantization methods are shown in Fig. 5. The average sidelobe levels obtained for the cases OWQ, OQW, and OQWS are respectively -48.4, -56.7 and -55.8 dB. Note that the branch and found method does approximately 8 dB better than the OWQ method for this case. Also note that the performance of the symmetric weights is not quite as good as the nonsymmetric weights. In order to compare the various waveforms in terms of a more well known quantity, consider the following function:

$$Q(\nu) = \sum_{k=k_1}^{k_2} S(k,\nu)$$

where $S(k,\nu)$ is defined in Sec. II. $Q(\nu)$, the Q function, is the clutter rejection obtained by using the burst waveform on clutter at doppler extending from $(K_1 - 1/2)T$ to $(K_2 + 1/2)T$ as compared to the rejection from a single pulse. $Q(\nu)$ is plotted in Fig. 6. The figure shows that the OQW and the OQWS give over 5 dB more clutter rejection over the doppler region of interest than does the OWQ.

ACKNOWLEDGEMENT

I would like to acknowledge the efforts of Steve Rainbolt who implemented the branch and bound algorithm.

REFERENCES

1. A. W. Rihaczak, Principles of High-Resolution Radar, McGraw Hill, New York (1969).

2. W. D. Rummler, "A Technique for Improving the Clutter Performance of Coherent Pulse Train Signals," IEEE Trans. Aero. Elec. Sys., Vol. AES-3, pp. 898-906 (November 1967).

3. D. F. Delong, "Design of Radar Signals and Receivers Subject to Implementation Errors," IEEE Trans. Info. Thy., Vol. IT-16, No. 6 (November 1970).

4. A. I. Cohen, "An Algorithm for Designing a Burst Waveform with Quantized Transmitter Weights," to appear.

5. R. Garfinkel and G. Neurhauser, Integer Programming, John Wiley and Sons, New York (1972).

6. L. J. Spafford, "Optimum Radar Signal Processing in Clutter," IEEE Trans. Info. Thy., Vol. IT-14, No. 5, pp. 734-743 (September 1968).

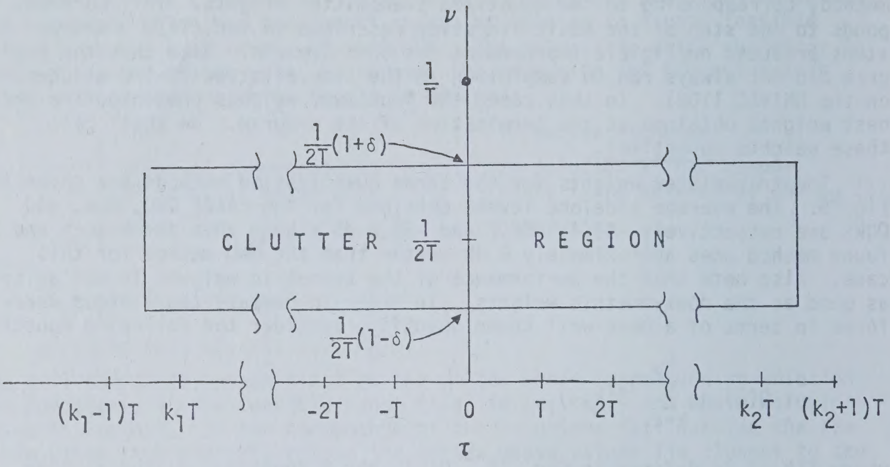

ν
1/T
1/2T(1+δ)
CLUTTER
1/2T
REGION
1/2T(1-δ)
(k1-1)T
k1T
-2T
-T
0
T
2T
k2T
(k2+1)T
τ

FIGURE 1 ASSUMED CLUTTER REGION

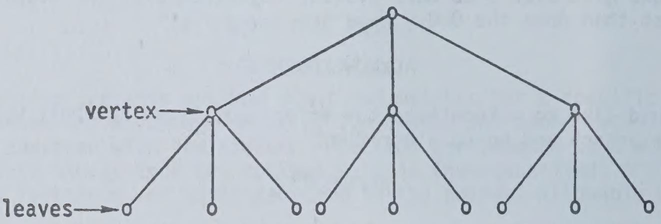

vertex
leaves

FIGURE 2 BASIC TREE STRUCTURE

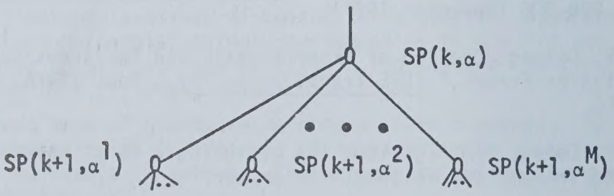

SP(k,α)
SP(k+1,α1)
SP(k+1,α2)
SP(k+1,αM)

FIGURE 3 BRANCHING FROM A VERTEX

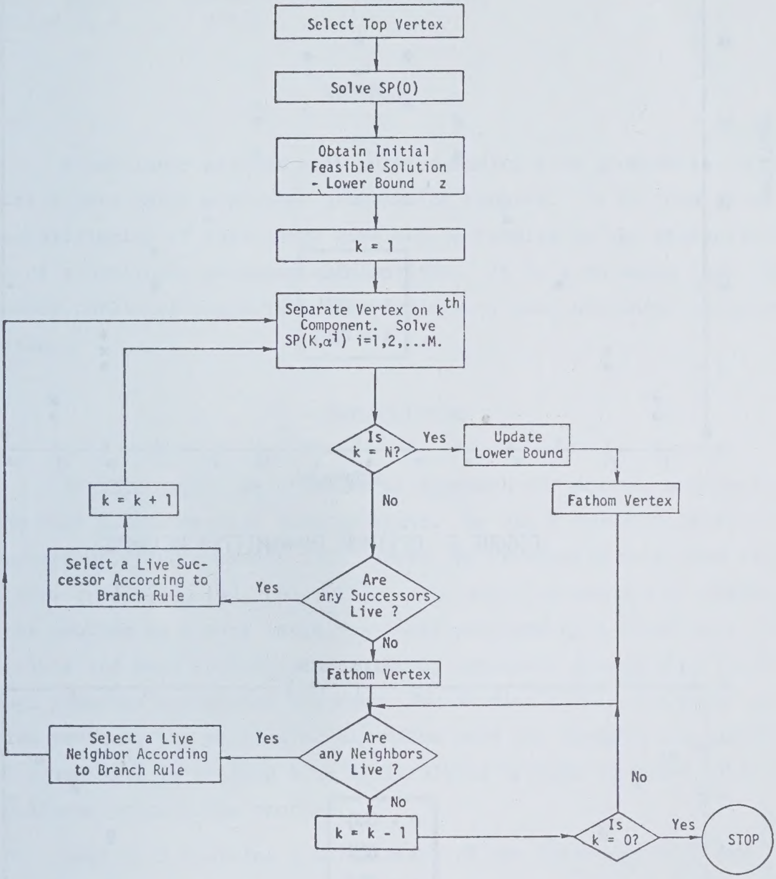

Select Top Vertex
Solve SP(0)
Obtain Initial Feasible Solution - Lower Bound z
k = 1
Separate Vertex on k^th Component. Solve SP(K,α^i) i=1,2,...M.
Is k = N?
Yes
Update Lower Bound
Fathom Vertex
No
k = k + 1
Select a Live Successor According to Branch Rule
Yes
Are any Successors Live ?
No
Fathom Vertex
Select a Live Neighbor According to Branch Rule
Yes
Are any Neighbors Live ?
No
k = k - 1
No
Is k = 0?
Yes
STOP

FIGURE 4 FLOW CHART OF TREE SEARCH STRATEGY

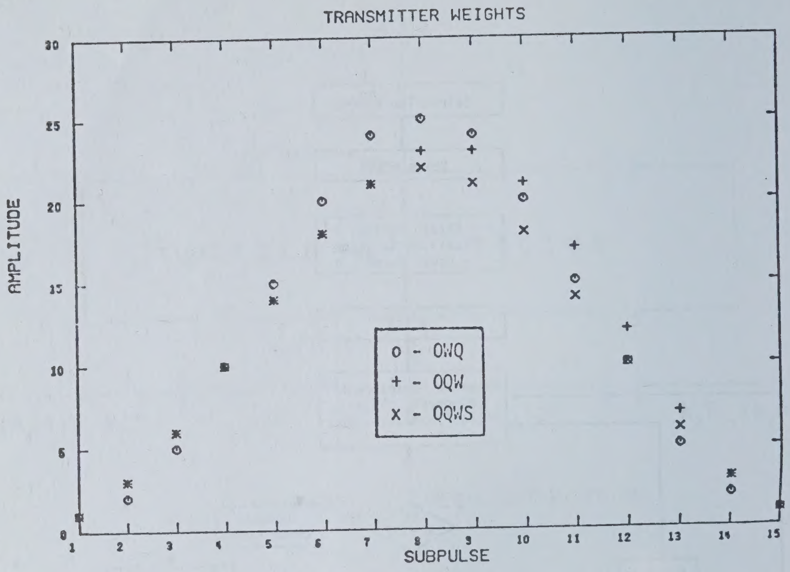


FIGURE 5 OPTIMUM TRANSMITTER WEIGHTS

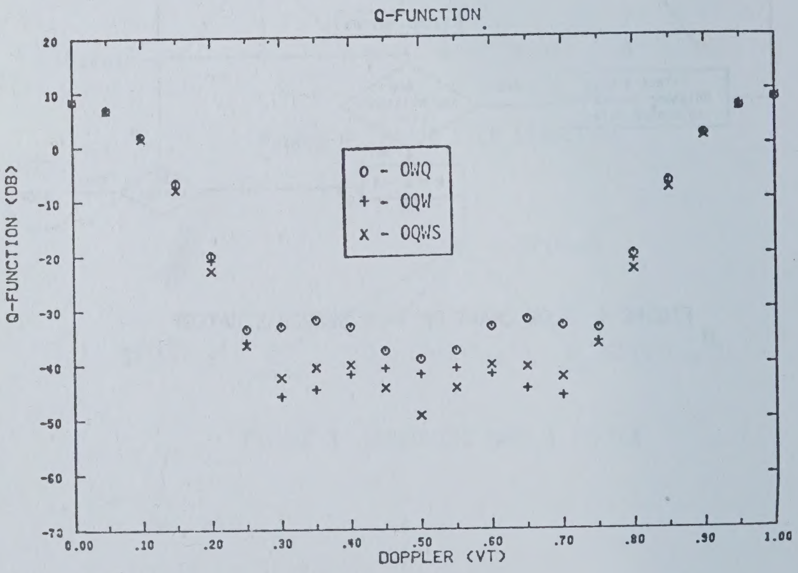


FIGURE 6 Q FUNCTION

SOLUTION TECHNIQUES FOR CERTAIN NONLINEAR MULTICOMMIDITY FLOW PROBLEMS

R. KLESSIG
Bell Telephone Laboratories, Incorporated
Holmdel, N. J. 07733

ABSTRACT

A nonlinear minimum cost multicommodity flow problem is formulated as a very large nonlinear programming problem. It is then shown how the difficulty of this large size can be handled by the appropriate choice of a nonlinear programming algorithm. It is also shown that certain previously published algorithms are versions of this nonlinear programming algorithm.

1 - INTRODUCTION

In this paper, we consider an approach for solving nonlinear minimum cost multicommodity flow problems. We use a somewhat different and perhaps unifying approach with respect to previously published solution techniques ([2], [3], [4], [8], [13]). The approach amounts to observing that the problem is a very large nonlinear programming problem with linear constraints and then applying a well known nonlinear programming algorithm. Although numerous algorithms are known for solving such a nonlinear programming problem, the particular algorithm used has significant computational advantages in dealing with difficulties arising from the size of the nonlinear programming problem.

Section 2 contains a formulation of the multicommodity flow problem while Section 3 has a description of the nonlinear programming algorithm and its properties. A discussion of the computational advantages and the relationship to other algorithms is contained in Section 4.

2 - PROBLEM FORMULATION

Consider an undirected graph with N nodes and M arcs. Each of L commodities must be routed from a given source node through the graph to a given sink node. Let

$d^\ell \triangleq$ positive flow requirement for commodity ℓ

$Q^\ell \triangleq$ number of paths without cycles from the source node to the sink node for commodity ℓ

$P_{\ell,j} \triangleq$ jth path without cycles from the source node to the sink node for commodity $\ell (j = 1,\ldots,Q^\ell)$

$b^k \triangleq$ initial nonnegative capacity of arc k

$u^k \triangleq$ nonnegative bound on the amount of capacity augmentation on arc k

$x_{\ell,j} \triangleq$ flow of commodity ℓ on path $P_{\ell,j}$

$z^k \triangleq$ nonnegative amount of capacity augmentation on arc k.*

With the above definitions, we can write the capacity constraint for each arc as

$$\sum_{\ell=1}^{L} \sum_{j=1}^{Q^\ell} a_{k,\ell,j} x_{\ell,j} - z^k \leq b^k \quad k = 1,\ldots,M \tag{2.1}$$

where

$$a_{k,\ell,j} = \begin{cases} 1 & \text{if arc } k \text{ is in path } P_{\ell,j} \\ 0 & \text{otherwise .} \end{cases} \tag{2.2}$$

The flow requirement constraint for each commodity is

$$\sum_{j=1}^{Q^\ell} x_{\ell,j} = d^\ell \quad \ell = 1,\ldots,L . \tag{2.3}$$

Finally, the decision variables are bounded.

$$0 \leq z^k \leq u^k \quad k = 1,\ldots,M \tag{2.4}$$

$$0 \leq x_{\ell,j} \qquad j = 1,\ldots,Q^\ell, \quad \ell = 1,\ldots,L . \tag{2.5}$$

---

* The idea of including capacity augmentation is due to R. A. Skoog.[11]

In order to simplify notation, let

$$x \triangleq [x_{1,1},x_{1,2},\ldots,x_{L,Q^L}]^T \in R^n \quad \left(n = \sum_{\ell=1}^{L} Q^{\ell}\right) \tag{2.6}$$

$$z \triangleq [z^1,\ldots,z^M]^T \in R^M \tag{2.7}$$

$$b \triangleq [b^1,\ldots,b^M]^T \in R^M \tag{2.8}$$

$$u \triangleq [u^1,\ldots,u^M]^T \in R^M \tag{2.9}$$

$$d \triangleq [d^1,\ldots,d^L]^T \in R^L . \tag{2.10}$$

Now, the appropriate matrices $A \in R^{M\times n}$ and $B \in R^{L\times n}$ can be constructed so that (2.1) through (2.5) can be compactly written as

$$Ax-z \leq b \tag{2.11}$$

$$Bx = d \tag{2.12}$$

$$0 \leq z \leq u \tag{2.13}$$

$$0 \leq x . \tag{2.14}$$

The function to be minimized over the constraint set defined by (2.11) through (2.14) is ϕ: $R^n \times R^M \to R^1$. ϕ is assumed to be of the form $\phi(x,z) = \psi(Gx,z)$ where ψ: $R^M \times R^M \to R^1$ is continuously differentiable and $G \in R^{M\times n}$ effects the linear transformation $v = Gx$ where

$$v^k = \sum_{\ell=1}^{L} \sum_{j=1}^{Q^{\ell}} g_{k,\ell}\, a_{k,\ell,j}\, x_{\ell,j} \qquad k = 1,\ldots,M \tag{2.15}$$

and $g_{k,\ell} > 0 \quad \forall k,\ell$. In words, the cost is a function of the weighted sum of the flows on each arc and the augmentation of capacity on each arc. Thus, the nonlinear minimum cost multicommodity flow problem can be written as

$$\min_{x,z}\{\psi(Gx,z)\,|\,Ax-z \leq b,\ Bx = d,\ 0 \leq x,\ 0 \leq z \leq u\}. \tag{2.16}$$

Problem (2.16) is a generalization of the problems considered in [2], [4], [8], and [13]. Furthermore, it is straightforward to generalize the problem and the resulting development in this paper by assuming each $g_{k,\ell}$ in (2.15) is a vector and hence each v^k is a vector. Problem (2.16) would then generalize the problem considered in [3].

3 - A NONLINEAR PROGRAMMING ALGORITHM

As a nonlinear programming problem, (2.16) has a rather pleasant structure. The cost function is continuously differentiable and the constraints are linear. Furthermore, it is easy to see that the constraint set is compact. Consequently, there are numerous algorithms from the nonlinear programming literature that can be applied to (2.16) to at least find points satisfying a necessary condition of optimality.

Unfortunately, except for the simplest of graphs, most of these algorithms will be impossible to implement. This is because the number of paths without cycles for each commodity will be extremely large for most interesting problems and thus it will be very difficult if not impossible to specify x in (2.16). In other words, the problem will defy formulation. This difficulty has been confronted in the study of linear multicommodity flow problems and some progress has been made in resolving it. Hence, the nonlinear programming algorithm selected for application to (2.16) should be such that it can make use of the results for the linear multicommodity flow problem. Such an algorithm is the Frank and Wolfe algorithm[7] (also called the Conditional Gradient Method[5]). The remainder of this section is a description of a version of this algorithm while a discussion of how it leads to linear multicommodity flow problems is deferred until Section 4.

The generic problem we are considering is

$$\min\{f(\omega)\,|\,\omega \in \Omega\} \tag{3.1}$$

where $f\colon\ R^p \to R^1$ is continuously differentiable and $\Omega \neq \phi$ is compact and convex. The algorithm can be viewed as evolving from the following optimality condition.

Proposition ([5], p. 91): (i) If $\hat{\omega}$ solves (3.1), then $\min\{<\nabla f(\hat{\omega}),\omega-\hat{\omega}>|\omega \in \Omega\} = 0$.* (ii) If f is convex and $\hat{\omega} \in \Omega$ satisfies $\min\{<\nabla f(\hat{\omega}),\omega-\hat{\omega}>|\omega \in \Omega\} = 0$, then $\hat{\omega}$ solves (3.1).☐ (3.2)

Algorithm: (3.3)

Step 0: Find any $\omega_0 \in \Omega$. Select parameters $0 < \alpha < 1$ and $0 < \beta < 1$. Set $i = 0$.

Step 1: Compute any $\tilde{\omega}_i \in \Omega$ satisfying

$$<\nabla f(\omega_i),\tilde{\omega}_i> = \min\{< \nabla f(\omega_i),\omega>|\omega \in \Omega\} .$$

Step 2: If $<\nabla f(\omega_i),\tilde{\omega}_i-\omega_i> = 0$, stop, ω_i satisfies the necessary condition of optimality in (3.2)(i); else compute the smallest nonnegative integer k_i such that

$$f\left(\omega_i + \beta^{k_i}(\tilde{\omega}_i-\omega_i)\right) - f(\omega_i) - \alpha\beta^{k_i}<\nabla f(\omega_i), \tilde{\omega}_i-\omega_i> \leq 0 . \quad (3.4)$$

Step 3: Set $\omega_{i+1} = \omega_i + \beta^{k_i}(\tilde{\omega}_i-\omega_i)$.

Step 4: Set $i = i+1$ and go to Step 1.☐

It should be noted that the algorithm has been modified to use the step size computation of Armijo[1] in Step 2.

The following theorem which states the convergence properties of algorithm (3.3) can be proved using (1.3.10) in [10].

Theorem: Let $\{\omega_i\}$ be the (possibly finite) sequence constructed by algorithm (3.3). Let (3.5)

$$\Delta \triangleq \left\{\omega' \in \Omega \middle| \min_{\omega\in\Omega} < \nabla f(\omega'),\omega-\omega'> = 0\right\}$$ = the set of all points

satisfying the necessary condition of optimality given by (3.2)(i). If $\{\omega_i\}$ is finite with last element ω_q, then $\omega_q \in \Delta$. If $\{\omega_i\}$ is infinite, then $d(\omega_i,\Delta)^\dagger \to 0$ as $i \to \infty$.☐

---

* ∇f is the gradient of f and $<\cdot,\cdot>$ is the Euclidean inner product.

† $d(\omega,\Delta) \triangleq \inf\left\{||\omega-\xi|| \middle| \xi \in \Delta\right\}$.

Corollary: Let $\{\omega_i\}$ be the (possibly finite) sequence constructed by algorithm (3.3). (i) If f is convex, then Δ can be replaced by $\hat{\Delta} \triangleq$ {all solutions to (3.1)} in Theorem (3.5). (ii) If f is strictly convex, then $\{\omega_i\}$ either terminates at or converges to the unique solution of (3.1). $\square$ (3.6)

4 - COMPUTATIONAL ADVANTAGES

The basic advantage of using algorithm (3.3) to solve (2.16) is that a sequence of linear multicommodity flow problems results. The advantages of this are discussed in this section.

Consider the optimization problem in Step 1 of algorithm (3.3). For (2.16), this is the linear programming problem

$$\min \langle G^T \nabla_v \psi(v_i, z_i), x\rangle + \langle \nabla_z \psi(v_i, z_i), z\rangle \tag{4.1}$$

s.t.

$$Bx = d$$

$$Ax - z \leq b$$

$$0 \leq z \leq u$$

$$0 \leq x$$

where $v_i = Gx_i$.*

This is a linear program that is closely related to the linear multicommodity flow problem considered in [12]. As a result, the column generation approach ([6],[12]) can be used in its solution. Suppose $\mu \in R^L$ and $\pi \in R^M$ are the simplex multipliers corresponding to the constraints $Bx = d$ and $Ax - z \leq b$ respectively at some iteration in the simplex algorithm. Let

$$[c^1, \ldots, c^M]^T = \nabla_v \psi(v_i, z_i) . \tag{4.2}$$

Then, a straightforward calculation shows that $x_{\ell,j}$ can enter the basis if

---

* Throughout this section, subscript i refers to an iteration index and not a vector component index.

$$\sum_{k=1}^{M}\left(c^k g_{k,\ell}a_{k,\ell,j} - \pi^k a_{k,\ell,j}\right) - \mu^\ell \tag{4.3}$$

$$= \sum_{k=1}^{M}\left(c^k g_{k,\ell} - \pi^k\right)a_{k,\ell,j} - \mu^\ell < 0 \quad .$$

Because of (2.2), the summation in (4.3) is just the length of path $P_{\ell,j}$ when each arc has length $c^k g_{k,\ell} - \pi^k$ $k = 1,\ldots,M$. Consequently, by carrying out a shortest path algorithm for each commodity, an appropriate $x_{\ell,j}$ variable (if any) can be found to enter the basis. Thus, all paths do not have to be prespecified but can be constructed as needed. Furthermore, no more than 2M+L paths will ever be needed to specify a solution. Finally, it should be noted that the structure of (4.1) can be exploited in its solution by the use of Generalized Upper Bounding.[9] Consequently, because the subproblem in Step 1 of algorithm (3.3) is a linear multicommodity flow problem, the difficulty in formulation for (2.16) can be avoided.*

If we denote the solution to (4.1) as $(\tilde{x}_i,\tilde{z}_i)$, we know that no more than 2M+L components of $(\tilde{x}_i,\tilde{z}_i)$ will be nonzero. Unfortunately, the same cannot be said of (x_i,z_i) since it is a linear combination of (x_0,z_0), $(\tilde{x}_1,\tilde{z}_1),\ldots, (\tilde{x}_{i-1},\tilde{z}_{i-1})$. In general, the number of nonzero components of (x_i,z_i) (and hence the memory required to specify it) will grow with i. (For f concave, this will not occur since $k_i = 0\ \forall i$ and hence $(x_i,z_i) = (\tilde{x}_{i-1},\tilde{z}_{i-1})\ \forall i$.) This can be avoided by using (v_i,z_i) as the iterate. If (v_i,z_i) is known, (4.1) is defined and hence $(\tilde{x}_i,\tilde{z}_i)$ can be found. Then, (3.4) becomes

$$\psi\left(v_i + \beta^{k_i}(G\tilde{x}_i-v_i),\ z_i + \beta^{k_i}(\tilde{z}_i-z_i)\right) - \psi(v_i,z_i) \tag{4.4}$$

$$- \alpha\beta^{k_i}\left[<\nabla_v\psi(v_i,z_i),G\tilde{x}_i - v_i> + <\nabla_z\psi(v_i,z_i),\tilde{z}_i-z_i>\right] \leq 0$$

and thus, $v_{i+1} = v_i + \beta^{k_i}(G\tilde{x}_i-v_i)$. Hence, (v_i,z_i) contains enough information to serve as the iterate and the memory required remains constant.

---

* Note, that to compute Gx, one needs only the values $g_{k,\ell}$. Hence, the very large G matrix need not be specified.

Of course, if this approach is followed, the information about the paths that each commodity follows is lost (since x_i is lost).

To conclude this section, we consider a special case of (2.16). This special case is such that algorithm (3.3) can be applied to yield a linear multicommodity flow at each iteration that is easier to solve than that discussed above. The special case is when all arcs have no existing flow capacity, all capacities can be augmented without bound and the cost does not decrease as the amount of flow capacity augmentation increases. Mathematically, we have

$$b = 0 \tag{4.5}$$

$$u = \infty \tag{4.6}$$

$$\nabla_z \psi(Gx,z) \geq 0 \quad \forall \ x,z \text{ feasible} . \tag{4.7}$$

It is easy to see that, if (4.7) holds, there is always an optimal solution to (2.16) such that equality holds in (2.11). Thus, we have the following

Proposition: If (4.5) through (4.7) hold and if $\tilde{x}$ solves (4.8)

$$\min\{\psi(Gx,Ax) | Bx = d, \quad x \geq 0\}, \tag{4.9}$$

then $(\tilde{x},A\tilde{x})$ solves (2.16). □

When algorithm (3.3) is applied to (4.9), the subproblem in Step 1 is

$$\min < \left[G^T \nabla_v \psi(v_i,z_i) + A^T \nabla_z \psi(v_i,z_i) \right], x> \tag{4.10}$$

s.t.

$$Bx = d$$

$$x \geq 0 ,$$

where $v_i = Gx_i$ and $z_i = Ax_i$. If we let

$$[s_{1,1},s_{1,2},\ldots,s_{L,Q^L}]^T = G^T \nabla_v \psi(v_i,z_i) + A^T \nabla_z \psi(v_i,z_i), \tag{4.11}$$

it follows that

$$s_{\ell,j} = \sum_{k=1}^{M} \left[g_{k,\ell} a_{k,\ell,j} \frac{\partial}{\partial v^k} \psi(v_i,z_i) + a_{k,\ell,j} \frac{\partial}{\partial z^k} \psi(v_i,z_i) \right] \tag{4.12}$$

$$= \sum_{k=1}^{M} \left[g_{k,\ell} \frac{\partial}{\partial v^k} \psi(v_i,z_i) + \frac{\partial}{\partial z^k} \psi(v_i,z_i) \right] a_{k,\ell,j} \, .$$

Thus, from the definition of $a_{k,\ell,j}$, $s_{\ell,j}$ is the length of path $P_{\ell,j}$ when each arc has length $g_{k,\ell} \frac{\partial}{\partial v^k} \psi(v_i,z_i) + \frac{\partial}{\partial z^k} \psi(v_i,z_i)$ $k = 1,\ldots,M$. Now (4.10) in terms of our original notation is

$$\min \sum_{j=1}^{Q^1} s_{1,j} x_{1,j} + \ldots + \sum_{j=1}^{Q^L} s_{L,j} x_{L,j} \tag{4.13}$$

s.t.

$$\sum_{j=1}^{Q^\ell} x_{\ell,j} = d^\ell \quad \ell = 1,\ldots,L$$

$$x_{\ell,j} \geq 0 \quad j = 1,\ldots,Q^\ell, \quad \ell = 1,\ldots,L \, .$$

But (4.13) can be solved by inspection and the solution is

$$\tilde{x}_{\ell,j} = \begin{cases} d^\ell & \text{if} \quad j = j(\ell) \\ 0 & \text{otherwise} \end{cases} \tag{4.14}$$

where $j(\ell)$ is any index satisfying

$$s_{\ell,j(\ell)} = \min \left\{ s_{\ell,j} \,\middle|\, j = 1,\ldots,Q^\ell \right\} . \tag{4.15}$$

In other words, (4.10) can be solved by simply routing each commodity on the shortest path when each arc has length $g_{k,\ell} \frac{\partial}{\partial v^k} \psi(v_i,z_i) + \frac{\partial}{\partial z^k} \psi(v_i,z_i)$. $k = 1,\ldots,M$. Algorithm (3.3) for this special case is essentially the same as the algorithms of [8] and [13].

5 - CONCLUSION

We have seen that the nonlinear minimum cost multicommodity flow problem is a nonlinear programming problem with simple structure but very many variables. By selecting the appropriate nonlinear programming algorithm, the difficulty with the large number of variables can be handled as in the case of linear multicommodity flow problems. Furthermore, for certain kinds of nonlinear minimum cost multicommodity flow problems, certain existing algorithms can be seen to be special cases of this nonlinear programming algorithm.

REFERENCES

[1] L. Armijo, "Minimization of Functions Having Lipschitz Continuous First Partial Derivatives," Pac. J. Math., 16(1), 1966, 1 through 3.

[2] S. C. Dafermos, "An Extended Traffic Assignment Model with Applications to 2-Way Traffic," Transportation Science, 5(4), 1971, 366 through 389.

[3] S. C. Dafermos, "The Traffic Assignment Problem for Multiclass-User Transportation Networks," Transportation Science, 6(1), 1972, 73 through 87.

[4] S. C. Dafermos and F. T. Sparrow, "The Traffic Assignment Problem for a General Network," J. Res. Nat. Bur. Standards - B. Math. Sciences, 73B, 1969, 91 through 118.

[5] V. F. Demyanov and A. M. Rubinov, Approximate Methods in Optimization Problems, American Elsevier, 1970.

[6] L. R. Ford, Jr., and D. R. Fulkerson, "A Suggested Computation for Maximal Multicommodity Network Flows," Management Science, 5, 1958, 97 through 101.

[7] M. Frank and P. Wolfe, "An Algorithm for Quadratic Programming," Nav. Res. Log. Quart., 3, 1956, 95 through 110.

[8] L. Fratta. M. Gerla and L. Kleinrock, "The Flow Deviation Method: An Approach to Store-and-Forward Communication Network Design," Networks, 3, 1973, 97 through 133.

[9] C. J. McCallum, Jr., "A Generalized Upper Bounding Approach to a Communications Network Flow Problem," VIII International Symposium on Mathematical Programming, Stanford University, August 27 through 31, 1973.

[10] E. Polak, Computational Methods in Optimization, Academic Press, 1971.

[11] R. A. Skoog, Bell Laboratories, Inc., Holmdel, N. J. 07733, private communication.

[12] J. A. Tomlin, "Minimum Cost Multicommodity Network Flows," Operations Research, 14, 1966, 45 through 51.

[13] B. Yaged, Jr., "Minimum Cost Routing for Static Network Models," Networks, 1, 1971, 139 through 172.

AN ALGORITHM FOR PLACING INTERCONNECTED OBJECTS IN A GIVEN CONFIGURATION AND ITS APPLICATIONS

R. R. Peiser
Schlumberger Well Services
P. O. Box 2175
Houston, Texas 77001

S. C. Lee
Department of Electrical Engineering
University of Houston
Houston, Texas 77004

ABSTRACT

The algorithm described in this paper is a method of mapping a set of connected objects into a given set of positions so that the cost of connecting these objects is either minimized or nearly minimized. In addition, several theorems have been presented in order to explain the basis of this algorithm. Although the computations required by the algorithm would be difficult for a human to perform, the technique is easily handled by a computer. The cost of having a machine find the mapping will be a fraction of the cost if performed by a human.

I. Introduction

The problem of efficiently placing interconnected objects is general problem that is encountered by several different disciplines. The problem is especially familiar to electronic circuit designers. After a circuit has been designed and made to work as a breadboard, next comes the tedious task of transforming the desing into a compact, connected configuration suitable for etching as a printed circuit board. More often than not, the size of the board is predetermined. The problem then is to find a configuration that will fit in the available space.

A great deal of effort has been spent on this problem. The task requires skill, experience, intuition, and time. All of these requirements are expensive. If an attempt to layout the board is unsuccessful after several weeks of effort, the layout is considered to be impossible, despite the fact that only a small fraction of the possible configurations have been tried.

Effort has also been spent to develop computer programs that perform the function of p.c. board layout. Several successful programs have been written [1,2,3]. The drawback to these programs is the fact that they are expensive to develop and use, are quite large, and they incorporate specific layout procedures that may not be acceptable for some users. In addition these programs still face the problem of finding a good initial layout.

The algorithm presented here is intended to provide either a man or machine the information necessary to make the first, and hopefully the final, layout. It is based on the fact that a good way to minimize the size of the layout is to start with a configuration that minimizes the lengths of the connecting pattern. Since these paths occupy space, both in the conductor and the insulating space around the conductor, a minimum path length configuration occupies minimum space. Consequently, if a minimum path configuration cannot be realized in the available space, one need not search for new configurations that may fit. Likewise, the probability of a successful layout is maximized when a minimum path layout is used.

II. Formulation of the Problem

There are several ways to construct a model for this problem. Graph theory techniques are applicable [4,5].

Several authors have found solutions to similar problems such as shortests paths within a graph [6,7] as well as algorithms for the shortest spanning subtree [8,9]. However, we have not found in literature the solution to this type of problem.

One approach is to consider two graphs with n vertices and weighted edges. One graph represents n connected objects and the other graph represents a configuration of n positions. The problem is to map one graph into the other with the constraint that the product of the edge weights is minimized. An exhaustive search of all possible mappings is not practical. For example, if there were 20 vertices (objects to be positioned) and a computer could automatically calculate the cost of a 20 position configuration every 500 microseconds, it would take about 40 million years to make the calculation and select the best configuration. Clearly, an algorithm must be found that avoids such a search.

The first step in formulating the problem is to define the manner in which the objects are connected. This can be done by listing the number of connections between each pair of objects. This will be the interconnection list. In general, this list will have $n(n-1)/2$ values.

The second step in formulating the problem is to define the given configuration. This can be done by listing the distance between all pairs of positions. This can be called the distance list. The assignment of a distance function can be somewhat arbitrary. For example, the distance between two positions may be defined as the distance between those objects along some admissable path. In addition, a penalty value may be added to a path that can cause path crossovers.

Let Γ be a set of functions that map the objects into the positions. If there are n objects and n positions, there will be n! possible mappings of the objects into the positions. These n! mappings are realizable mappings. That is, the mappings have certain properties: For a given configuration:

Property R_1	If an object maps into a particular position, then no other object may map into that position.
Property R_2	An object may map into one position only.
Property R_3	If p objects map into p positions, then the remaining n-p objects must map into the remaining n-p positions.

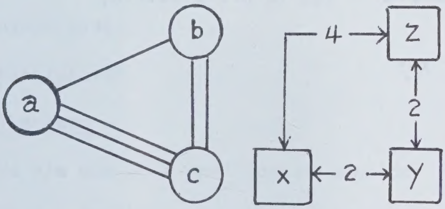
These rather obvious properties will be useful when we begin to calculate the cost of a configuration.

III. The Interconnection Matrix, the Distance Matrix and the Inequality Matrix

In order to simplify the procedure of calculating the cost (total path length) of a configuration, we will write the interconnection list as a matrix of dimension 1×J. This will be called the interconnection matrix $[C_1,C_2,C_3,\ldots,C_J]$. $[C]_R$ is the interconnection matrix when the values are placed in some order R. The distance list can also be written as a matrix of dimension 1×J. This will be the distance matrix $[D]_R = [D_1,D_2,D_3,\ldots,D_J]$. Note that if there are J elements of [C] and [D], there will be J! orderings in each matrix.

We can see then, that if C_1 is the number of connections between objects a and b and D_1 is the cost of a connection between positions x and y then, the product C_1D_1 is the cost of the connections between objects a and b when these objects are located in positions x and y.

The total cost of path connections for a given configuration is called X_R and is $[C]_R[D]_R^T$ and the elements of [C] and [D] are placed in an order that is a result of a relizable mapping. For example, consider three objects a,b, and c that map into positions x,y, and z respectively. Each value of C in the interconnection matrix represents the number of connections between two objects.



$C(a,b)=1$
$C(b,c)=2$
$C(a,c)=3$

$D(x,y)=2$
$D(y,z)=2$
$D(x,z)=4$

Then x, the cost of the mapping is

$$= [C(a,b) \quad C(a,c) \quad C(b,c)] \begin{bmatrix} D(x,y) \\ D(x,z) \\ D(y,z) \end{bmatrix} = [1,3,2] \begin{bmatrix} 2 \\ 4 \\ 2 \end{bmatrix} = 18$$

Conversely, if we know [C] and [D], we can sometimes generate a mapping.

For example if $[C]_R = [C(b,c) \quad C(a,c) \quad C(a,b)]$

and $[D]_R = [D(x,z) \quad D(y,z) \quad D(x,y)]$

then b maps to (x or z) and (x or y) b→x
c maps to (x or z) and (y or z) c→z
a maps to (x or y) and (y or z) a→y

Not all mappings are realizable from $[C]_R$ and $[D]_R$. For example,

let $[C]_R = [C(a,b) \quad C(a,c) \quad C(a,d) \quad C(b,c) \quad C(b,d) \quad C(c,d)]$

and $[D]_R = [D(y,z) \quad D(x,z) \quad D(x,y) \quad D(w,z) \quad D(w,y) \quad D(w,x)]$

In order to be realizable, (a) must map into (y or z) and (x or z) and (x or y). Clearly, no position satisfies the conditions of the mapping.

At this point, it will be convenient to define an inequality matrix [Q].

$$\text{where } [Q] = \begin{bmatrix} C_1D_1 & C_1D_2 & C_1D_3 & \cdots\cdots & C_1D_J \\ C_2D_1 & C_2D_2 & C_2D_3 & \cdots\cdots & C_2D_J \\ C_3D_1 & C_3D_2 & C_3D_3 & \cdots\cdots & C_3D_J \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ C_jD_1 & C_JD_2 & C_JD_3 & \cdots\cdots & C_JD_J \end{bmatrix}$$

and $C_1 \geq C_2 \geq C_3 \geq \ldots \geq C_J$

and $D_1 \geq D_2 \geq D_3 \geq \ldots \geq D_J$

This matrix is a square matrix of dimension J×J. In addition, whatever general properties that exist for [Q] also exist for $[Q]^T$.

Let L be the set of all numbers that represent a summation of elements C_iD_j such that each element in the summation is unique in each row and column in [Q]. Note that if [Q] is a matrix of dimension J×J, then there will be J! elements in L. For example if [Q] is a 2×2 matrix,

$$[Q] = \begin{matrix} AX & AY \\ BX & BY \end{matrix} \qquad \text{where } A \geq B,\ X \geq Y$$

then $L = \{AX+BY, AY+BX\} = \{\ell_1, \ell_2\}$

If [Q] is a 3×3 matrix,

$$[Q] = \begin{bmatrix} AX & BX & CX \\ AY & BY & CY \\ AZ & BZ & CZ \end{bmatrix} \qquad \begin{matrix} \text{Where } & A \geq B \geq C \\ \text{and } & X \geq Y \geq Z \end{matrix}$$

then $L = \{AX+BY+CZ,\ BX+CY+AZ,\ CX+AY+BZ,\ BX+AY+CZ,\ AX+CY+BZ,\ CX+BY+AZ\}$

$= \{\ell_1, \ell_2, \ell_3, \ell_4, \ell_5, \ell_6\}$

The elements of [Q] have certain interesting properties. For the 2×2 matrix, we find that

Property Q1 $\quad AX+BY \geq AY+BX$

The proof can be seen from the fact that

$A \geq B$ and $X \geq Y$ or $X-Y \geq 0$

Then $A(X-Y) \geq B(X-Y)$

and $AX+BY \geq AY+BX$

It can also be seen that

Property Q2 $\quad AX+BX \geq AY+BY$

because $X(A+B) \geq Y(A+B)$

and $X \geq Y$

In the 3×3 matrix where $A \geq B \geq C$ and $X \geq Y \geq Z$, we find that

Property Q3 $AX+BZ \geq BX+AZ$

Proof. Given that $X \geq Z$ and $A-B \geq 0$

then $X(A-B) \geq Z(A-B)$

or $AX+BZ \geq BX+AZ$

$$[Q] = \begin{bmatrix} \enclose{circle}{AX} & AY & \boxed{AZ} \\ \boxed{BX} & BY & \enclose{circle}{BZ} \\ CX & CY & CZ \end{bmatrix}$$

It can also be seen that $BX+CZ>BZ+CX$ from Property Q3. In addition, since the properties that hold for $[Q]$ also hold for $[Q]^T$, then

$AX+CY \geq AY+CZ$

and $AY+CZ \geq AZ+CY$

IV. Theorems

We are now in a position to consider the following theorems.

Theorem 1 The upper bound, ℓ_u, for all $\ell \varepsilon L$ for a J×J inequality matrix is

$$\sum_{i=1}^{J} C_i D_i$$

This upper bound for all $\ell \varepsilon L$ is the summation of the elements of Q that are in the diagonal from upper left to lower right.

The proof of this theorem can be found by first considering the 2×2 minor matrix on the lower right hand corner of [Q]. This matrix is

$$\begin{bmatrix} \enclose{roundedbox}{C_{J-1}\ D_{J-1}} & C_{J-1}\ D_J \\ C_J\ D_{J-1} & \enclose{roundedbox}{C_j\ D_J} \end{bmatrix}$$

From Property Q_1; we see that $C_{J-1}\ D_{J-1} + C_J D_J$ is the upper bound for this 2×2 minor matrix.

Using Property Q_2, we see that $C_{J-2}\ A_{J-2} + C_{J-1}\ D_{J-1} + C_J D_J$ is the upper bound for the 3×3 minor matrix. This is true because $C_{J-2} \geq C_{J-1} \geq C_J$ and $D_{J-2} \geq D_{J-1} \geq D_J$. The process is repeated until the $C_1 D_1$ term is added to this summation. This set of elements of [Q] lies along the main diagonal from upper left to lower right. This is called the upper bound diagonal U.B.D.

Theorem 2 The lower bound, ℓ_1, for all $\ell \varepsilon L$ in a J×J inequality matrix is

$$\sum_{i=1}^{J} C_i D_{J+1-i}$$

Note also that $\ell_\ell \varepsilon L$. This is called the lower bound diagonal L.B.D.

The development of properties of the inequality matrix and Theorem 1 and Tehorem 2 is pertinent to the problem at hand because the terms C_1, C_2, ... C_J can represent sorted values of the interconnection list. Similarly, the values D_1, D_2,...,D_J can represent sorted values in the adjacency list. For this condition,

$$J = \frac{n(n-1)}{2}$$

where n is the number of objects to be mapped.

Theorem 3 If the elements of an inequality matrix [Q] are comprised of elements of the interconnection and distance list, then X, the set of costs for all realizable configurations, is a subset of L. The proof can be seen from the definition of X and L. In general, then a n! elements of X and $[\frac{n(n-1)}{2}]!$ elements of L.

Theorem 4 Where an inequality matrix is comprised elements of the interconnection and distance lists for n=3, then L is equal to X.

Proof: We know from Theorem 3 that X is a subset of L. We also know that there are n(n-1)!/2 elements of L and n! elements of X. When n=3, n(n-1)/2 = n. Therefore, every element of X is an element of L, and L has no elements not in X, therefore X=L.

Corollary C1. If ℓ_u is the upper bound for the elements in L, then ℓ_u is an upper bound of X.

Corollary C2. If ℓ_ℓ is the lower bound for the elements in L, then ℓ_ℓ is a lower bound of X.

Corollary C3. For n=3, then ℓ_u the upper bound of L is equal to the upper bound of X and $\ell_u \varepsilon X$. For n=3, then ℓ_ℓ the lower bound of L is equal to the lower bound of X and $\ell_\ell \varepsilon X$.

The corollaries imply that for n=3 we can always find realizable mappings that will be qual to the upper and lower bound of the connection cost X. Futhermore, we have established upper and lower bounds for the elements of X for any $n \geq 3$. Knowledge of these bounds can be useful in evaluating the performance of a chosen configuration.

An example of how these findings can be used to find a good configuration is presented. Consider four objects a,b,c,d and four positions w,x,y,z.

Let the interconnection and distance lists be:

$C(a,b)=6=C_1$
$C(a,c)=5=C_2$
$C(a,d)=4=C_3$
$C(b,c)=3=C_4$
$C(b,d)=2=C_5$
$C(c,d)=1=C_6$

$D(w,x)=1=D_6$
$D(w,y)=2=D_5$
$D(w,z)=3=D_4$
$D(x,y)=4=D_3$
$D(x,z)=5=D_2$
$D(y,z)=6=D_1$

The upper bound

$$\ell_u = \sum_{i=1}^{6} C_i D_i = 91$$

The lower bound

$$\ell_\ell = \sum_{i=1}^{6} C_i D_{(7-i)} = 56$$

The next question is whether or not there is a realizable configuration that has 56 as its cost function. The check for realizability is as follows:

For a minimum cost configuration; C_1 is paired with D_6, C_2 with D_5, C_3 with D_4, C_4 with D_3, C_5 with D_2, and C_6 with D_1. Or

a,b map into w,x
a,c map into w,y
a,d map into w,z
b,c map into x,y
b,d map into x,z
c,d map into y,z

so object "a" maps into position (w or x) and (w or y) and (w or z). Consequently, a is mapped into w. Object "b" maps into positions (w or x) and (x or y) and (x or z). Note that if a maps to w and C_1 is paired with D_6, then b must map into position x. It can also be seen that the mapping of b into x is realizable. Similarly, C maps to y and d maps into z.

The cost of this configuration is 56, the lower bound of L.

The procedure, then, is to find a realizable configuration that has its paired terms positioned near the lower bound diagonal (L.B.D.) of the inedquality matrix. It is suggested that one find realizable mappings along the L.B.D. starting at its corners. So a and b map to y and z (C_1D_6 pairing); and c,d map into w,x (C_6D_1 pairing).

It is usually a good procedure to start the realization at the corners, i.e. pair C_1D_6 and pair C_6D_1. At least one of the corner values can always be realized, and any departures from the L.B.D. will occur in the center of [Q]. In fact, one can judge the effectiveness of a realization by examining the pairings in the inequality matrix.

Example 2

Consider a 4x4 inequality matrix, where $A \geq B \geq C \geq D$ and $w \geq x \geq y \geq z$.

$$\begin{bmatrix} AW & BW & CW & \enclose{circle}{DW} \\ AX & \enclose{circle}{BX} & CX & DX \\ AY & BY & \enclose{circle}{CY} & DY \\ \enclose{circle}{AZ} & BZ & CZ & DZ \end{bmatrix} \text{Realization 1} \qquad \begin{bmatrix} \enclose{circle}{AW} & BW & CW & DW \\ AX & BX & \enclose{circle}{CX} & DX \\ AY & \enclose{circle}{BY} & CY & DY \\ AZ & BZ & CZ & \enclose{circle}{DZ} \end{bmatrix} \text{Realization 2}$$

Although both realizations have two elements on the L.B.D. and two elements off of the L.B.D. Realization 1 has a lower cost function. This can be seen from the fact that the AW and DZ terms are greater departures from the L.B.D. than BX and CY.

Proof for Example 2

$$AW+BY+CX+DZ \geq AZ+BX+CY+DW$$

can be written as

$$(W-Z)(A-D) \geq (X-Y)(B-C)$$

so from Property Q_2

and $(W-Z) \geq (W-Y)(X-Y)$

$(A-D) \geq (A-C)(B-C)$

so $W-Z \geq X-Y$

$A-D \geq B-C$

Therefore, $AW+BY+CX+DZ \geq AZ+BX+CY+DW$

Up to now, we have been considering interconnections and adjacencies between pairs of objects. However, it may be useful to consider groupings of more than two objects. In particular, groups of three objects seem particularly interesting. We could describe the interconnection of ternary groups as the number of connections that exist among three objects.

The ternary grouping interconnection list can be generated from the interconnection list for pairs of objects. For example, the number of interconnections among objects p, q, and r is

$$C(p,q,r) = C(p,q) + C(p,r) + C(q,r)$$

The same procedure can be used to calculate the adjacency among ternary groups. The number of elements in an interconnection list for n objects with groupings of g objects is $n!/g!(n-g)!$. For ternary groups, then, the number of elements in an interconnection list is $K = n(n-1)(n-2)/6$. The adjacency list also has K elements.

The values of the interconnection and adjacency lists for ternary groups can also be sorted and placed in an inequality matrix. The matrix will now be of dimension K×K; and of course, the properties for the inequality matrix are the same as before.

V. An Algorithm

The previous discussion has shown that a realizable configuration with minimum cost is easily found from the inequality matrix for $n=3$. However, the problem becomes extremely complicated in larger values of n. The upper and lower bounds of the values of L can be useful in evaluating the performance of a particular mapping, but these bounds do not provide insight as to how a "good" mapping can be found. Consequently, other facets of the information derived thus far must be used to find a useful algorithm.

First we know that a realizable configuration with a minimum cost can always be found when the number of objects is 3. Consequently, we know that if a problem can be reduced to a three object problem, a solution can be found.

Secondly, we know that the values at the corner of the inequality matrix should be realized first. We also know that from the conditions of realizability, that once a decision has been made to map an object into a certain position, that mapping must hold true in all values in the inequality matrix. Using this information, an algorithm is suggested.

Step 1 Generate the ternary grouping interconnections and distance lists.

Step 2 Search the interconnection list for the maximum number of connections. Note the objects.

Step 3 Search the distance lists for the minimum number of connections. Note the positions.

Step 4 Assign those three objects to those three positions so that the cost of that assignment is minimized.

We now must map the remaining objects into the remaining positions. Note, however, that we have only n-3 objects to map. The process is repeated until there are no more ternary groups left to map. This procedure uses only one corner of the inequality matrix, but it avoids the problem of checking for realizability. One corner of the inequality matrix is always realizable.

After the ternary groups have been mapped, a remainder group must be mapped. If there is only one remaining object, then it will be mapped into the remaining position. If there are two objects to be mapped, the total cost of each mapping can be calculated, and the minimum cost mapping can be chosen.

Applying this algorithm to example 1, we generate four ternary groups of objects and generate their connection values.

$$\begin{aligned} C(abc) &= C(ab) + C(ac) + C(bc) = 14 \\ C(abd) &= C(ab) + C(bd) + C(ad) = 12 \\ C(acd) &= C(ac) + C(ad) + C(cd) = 10 \\ C(bcd) &= C(bc) + C(cd) + C(bd) = 6 \end{aligned}$$

For the positions:

$$\begin{aligned} D(wxy) &= D(wx) + D(wy) + D(xy) = 7 \\ D(xyz) &= D(xy) + D(xz) + D(yz) = 15 \\ D(wxz) &= D(wx) + D(wz) + D(xz) = 9 \\ D(wyz) &= D(wy) + D(wz) + D(yz) = 11 \end{aligned}$$

So, objects abc (with C=14) map into wxy (with D=7), and object d, then, maps into z. Then a→w, b→x, c→y for the minimum cost of that ternary group.

VI. Conclusion

A computer program using this algorithm has been prepared to solve problems with up to 21 objects. The execution time is well under one minute. The output data from the program are the upper and lower bounds of L, the mapping selected, and the total path length that mapping. The results of the program for various types of input data were generally good, but not always optimum. It is felt that the algorithm could be improved by using the inequality matrix to a greater extent. In particular, both corners of the inequality matrix should be used to find realizable low cost mappings. The program is now being modified to incorporate this feature.

References

1. C. J. Fisk, D. L. Caskey, and L. E. West, "ACCEL: Automated Circuit Card Etching Layout," Proceedings of IEEE, Vol. 55, pp. 1971-1982, November 1967.

2. Arnold Spitalny, "On-line Graphics Applied to Layout Design of Integrated Circuits," Proceedings of the IEEE, Vol. 55, pp. 1982-1988, November 1967.

3. J. F. Jamison, "Programmers Manual: UCARDS Union Carbide Automatic Routing and Design System for Printed Circuit Boards," Union Carbide Corporation Nuclear Division, April 30, 1969.

4. N. Deo, Graph Theory with Applications to Engineering and Computer Science, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1974.

5. F., Harary, Graph Theory, Addison-Wesley Publishing Co., Reading, Mass., 1969.

6. J. Boothroyd, "Algorithm 23: Shortest Path Between Start Node and All Other Nodes in a Network," The Computer Journal, Vol. 10, No. 3, Nov. 1967, 307-308.

7. Y. Tabourier, "All Shortest Distances in a Graph. An Improvement to Dantzig's Inductive Algorithm," Discrete Mathematics, Vol. 4, No. 1, Jan. 1975, 83-87.

8. J. B. Krushal, Jr., "On the Shortest Spanning Subtree of Graph on the Traveling Salesman Problem," Proc. Am. Math. Soc., Vol. 7, 1956, 48-50.

9. R. C. Prim, "Shortest Connection Networks and Some Generalizations," Bell System Tech. J., Vol. 36, Nov. 1957, 1389-1401.

CONTROLLABILITY, POLE PLACEMENT AND STABILIZABILITY IN LARGE SCALE COMPOSITE SYSTEMS[†]

U. OZGUNER and W. R. PERKINS
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois 61801

ABSTRACT

There exist many physical systems, e.g. power systems, economic systems, neurological systems, that are composed of a number of interconnected subsystems. This paper proposes a Large Scale Composite (LSC) system model for their analysis and control. Each subsystem is assumed to be linear, time-invariant and have both interaction inputs from other subsystems and local inputs. Results on controllability, pole placement and stabilizability using local feedback are given for the general LSC system and for the special case of a chain of subsystems in tandem.

I. INTRODUCTION

Many systems of interest, including power systems, economic systems, and neurological systems, exhibit a complex structure of interconnected subsystems. In this paper a mathematical representation of such systems composed of interconnections of many subsystems is proposed. This representation is called the Large Scale Composite (LSC) System. Each subsystem is assumed to be linear, time invariant and have both interaction inputs from the other subsystems and local inputs.

Alternative models of large scale systems have been proposed. In particular, decentralized system structures have been considered [1,2,3]. As will be shown below, additional insight into large scale system structures can be achieved using the LSC model that is not available with decentralized models.

II. CONTROLLABILITY OF SUBSYSTEMS

The system S is composed of m subsystems $s_1, s_2, \ldots, s_m$. The j^{th} subsystem will be modeled by

---

[†] This work was supported in part by the Joint Services Electronics Program (U. S. Army, U. S. Navy, and U. S. Air Force) under Contract DAAB-07-72-C-0259 and in part by the U. S. Air Force under Grant AFOSR-73-2570.

$$s_j: \quad \dot{x}^j = A^j x^j + \sum_{i=1}^{m} B_i^j u_i^j$$
$$y^j = C^j x^j + \sum_{i=1}^{m} D_i^j u_i^j \tag{1}$$

where u_i^j, for $i \neq j$, are the interaction inputs, i.e. input from other subsystems, and are defined as,

$$u_i^j = h_{ij} y^i \quad \text{for} \quad i \neq j. \tag{2}$$

Moreover, x^j, u_i^j, y^j are n_j, r_{ij} and q_j vectors respectively, and the matrices are real and of proper dimension. The input of s_j which is not connected to any other subsystem is denoted by u_j^j, and is called "the locally available input."

For notational simplicity in this section the index j in (1) is dropped, and we define $B_2 = (B_1^j, B_2^j, \ldots, B_{j-1}^j, B_{j+1}^j, \ldots, B_m^j)$ and $B_1 = B_j^j$. D_2, D_1, u_2 and u_1 are formed similarly to give

$$\dot{x} = Ax + B_1 u_1 + B_2 u_2$$
$$y = Cx + D_1 u_1 + D_2 u_2 \tag{3}$$

as the equations for any one subsystem. Here u_1 shows the locally available inputs and u_2 the interaction inputs, i.e. all inputs from other subsystems.

A canonical form is now discussed that exhibits transparently the controllability properties of the subsystem.

Theorem 1: There exists a linear transformation $x(t) = Tx'(t)$ such that (3) is transformed into the controllability decomposition canonic form

$$\begin{bmatrix} \dot{x}_1' \\ \dot{x}_2' \\ \dot{x}_3' \\ \dot{x}_4' \end{bmatrix} = \begin{bmatrix} A_{11} & 0 & 0 & A_{14} \\ 0 & A_{22} & 0 & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ 0 & 0 & 0 & A_{44} \end{bmatrix} \begin{bmatrix} x_1' \\ x_2' \\ x_3' \\ x_4' \end{bmatrix} + \begin{bmatrix} B_{11} \\ 0 \\ B_{31} \\ 0 \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ B_{22} \\ B_{32} \\ 0 \end{bmatrix} u_2 \tag{4}$$

where

$$[A_{11}, B_{11}], [A_{22}, B_{22}], \left[\begin{bmatrix} A_{11} & 0 \\ A_{31} & A_{33} \end{bmatrix}, \begin{bmatrix} B_{11} \\ B_{31} \end{bmatrix} \right] \text{ and } \left[\begin{bmatrix} A_{22} & 0 \\ A_{32} & A_{33} \end{bmatrix}, \begin{bmatrix} B_{22} \\ B_{32} \end{bmatrix} \right]$$

are controllable pairs.

For a proof and an algorithm to construct T, see [4]. When the subsystem is controllable (4) would be,

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} A_{11} & 0 & 0 \\ 0 & A_{22} & 0 \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} B_{11} \\ 0 \\ B_{31} \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ B_{22} \\ B_{32} \end{bmatrix} u_2 . \qquad (5)^{\dagger}$$

Consider the local linear state feedback:

$$u_1 = [K_1 \; K_2 \; K_3] \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} . \qquad (6)$$

It can be shown [4] that the following observations hold:

1. Eigenvalues related to A_{11} can be arbitrarily placed by local linear state feedback, without any constraints because of interconnections.

2. Eigenvalues related to A_{22} cannot be shifted by local feedback.

3. If there is no feedback through the interconnections the LSC system is locally stabilizable by local linear state feedback if and only if eigenvalues of A_{22} are in the left half complex plane (for every subsystem).

4. If there is no feedback through the interconnections eigenvalues related to A_{33} can be placed by local feedback.

III. PREDECESSORS IN LSC SYSTEMS

Having looked at the subsystems at the local level, we can now address the question of overall characteristics in more detail. For this, some definitions and some new concepts will be used.

Definition 1: Let $\{s_1,s_2,\ldots,s_m\}$ be the subsystems of a LSC system S. s_i is a predecessor of s_j if the output of s_i appears as a component of the input to s_j. That is, s_i is a predecessor of s_j if $h_{ij} \neq 0$.

[†] Aoki has previously used this model in a different context [1,2]. He has considered the above as the complete model for a large scale system with two "control agents." An extension of the decomposition for m control agents would be fairly complicated requiring all pairwise controllable subspaces, triply controllable subspaces etc. Note also that Aoki's results are not directly applicable since the inputs u_2 are not locally available in our model.

In any system sets of inputs which are necessary for controllability may be identified [5]. Similarly, in LSC systems we would be interested in the class of necessary predecessors for controlling s_j. Denote such a class by P(j). Each $p \in P(j)$ shows a set of numbers identifying the predecessors. Denote the set of predecessors of s_j by K(j). For all $p \in P(j)$, $p \subset K(j)$.

Definition 2: The predecessor composite system (PCS) is the quadruple $\{\bar{A}^j, \bar{B}^j, \bar{C}^j, \bar{D}^j\}$, where

$$\bar{C}^j = \text{block diag.}\ (\ldots, h_{kj}C^k, \ldots) \quad \text{for all } k \in K(j)$$

$$\bar{D}^j = \text{block diag.}\ (\ldots, h_{kj}D^k, \ldots) \quad \text{"} \quad \text{"} \quad \text{"}$$

$$\bar{A}^j = \text{block diag.}\ (\ldots, A^k, \ldots) \quad \text{"} \quad \text{"} \quad \text{"}$$

$$\bar{B}^j = \text{block diag.}\ (\ldots, B^k, \ldots) \quad \text{"} \quad \text{"} \quad \text{"}$$

It is obvious that the output of the PCS is the interaction input of s_j. The following theorem is self-evident:

Theorem 2: A necessary condition for controllability of a subsystem s_j is that for at least one $p \in P(j)$ its PCS is output controllable.

p actually shows a subset of the PCS's outputs and may be determined using an algorithm given in [4].

The PCS will also have local and interaction inputs, and a question is whether its local inputs are sufficient for output controllability. Let the subindex l denote local inputs, and the subindex p denote outputs related to $p \in P(j)$ in the PCS. Let $\bar{A}^j$ be $\bar{n}(j) \times \bar{n}(j)$ and p have r_p^j elements. For local inputs of the PCS to be sufficient for controllability of s_j,

$$\text{rank}[\bar{C}_p^j\bar{B}_1^j \,\vdots\, \bar{C}_p^j\bar{A}^j\bar{B}_1^j \,\vdots\, \cdots \,\vdots\, \bar{C}_p^j[\bar{A}^j]^{\bar{n}(j)-1}\,\bar{B}_1^j \,\vdots\, \bar{D}_{1p}^j] = r_p^j \,. \qquad (7)$$

If the rank of the matrix in (7) turns out to be less than r_p^j then the space whose dimension is equal to the difference must be controlled through the interaction inputs of the PCS. An extension of the controllability decomposition canonic form is possible to similarly decompose the output-controllable space of the PCS. Thus, it will be possible to identify the interaction inputs of the PCS necessary to augment $\bar{B}_1^j$, so that (7) holds. This may lead to a sequence of steps backwards along the interaction inputs of PCS's. Such a sequence must end at a local input as the following example will demonstrate.

Example 1: Consider the three interconnected subsystems,

$$
\begin{aligned}
\dot{x}^1 &= A^1x^1 + B^1_1u^1_1 + B^1_2u^1_2 + B^1_4u^1_4 \\
\dot{x}^2 &= A^2x^2 + B^2_2u^2_2 + B^2_3u^2_3 \qquad (8) \\
\dot{x}^3 &= A^3x^3 + B^3_3u^3_3 + B^3_1u^3_1
\end{aligned}
$$

where

$$
A^1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} \qquad A^2 = \begin{bmatrix} 4 & 0 \\ 0 & 5 \end{bmatrix} \qquad A^3 = \begin{bmatrix} 6 & 0 \\ 0 & 7 \end{bmatrix}
$$

$$
B^1_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad B^1_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad B^1_4 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad B^2_2 = B^3_3 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad B^2_3 = B^3_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}
$$

$$
y^1 = x^1_2 \qquad y^2 = x^2_2 \qquad y^3 = x^3_2 \qquad h_{21} = h_{32} = h_{13} = 1
$$

and where u^1_4 shows a connection with other subsystems. Each subsystem is completely state controllable and output controllable using all its inputs. However when considering local inputs as separate from interaction inputs it can be seen that the output controllable spaces left for interaction inputs to control form a loop.†

Theorem 3: A further necessary condition for controllability of s subsystem s_j, when Theorem 2 holds, is that it is locally controllable by its PCS, or the deficiency is controllable by the preceding PCS, or this sequence end at local inputs in a finite number of steps.

IV. THE MULTILEVEL AND CHAIN STRUCTURES

Definition 3: If the LSC system $\{s_1, s_2, \ldots, s_m\}$ can be decomposed into disjoint subsets, such that there is no loop among the subsets, the LSC system is said to have a multilevel structure. The subsets of subsystems are the constituents of this structure.

The multilevel structure is important since the controllability conditions will have to be checked for each constituent separately instead of for the whole LSC system. (For an algorithm using graph theory to identify the constituents and further implications of the multilevel structure, see [6].)

---

† The term loop has been used in a literal sense, though a graph theoretic interpretation and analysis of the interconnections in a LSC system is always possible. See [6].

Definition 4: The special multilevel structure composed of m subsystems connected in tandem is called a chain.

We shall now analyze the controllability of chains. It is clear that the results can be extended to multilevel LSC systems where each constituent is a single subsystem or where the subsystems within each constituent can be modeled as a single system [6].

First consider a chain of m controllable subsystems with no local inputs. Let the systems be:

$$s_1: \quad \begin{aligned} x^1 &= A^1x^1 + B^1u^1 \\ y^1 &= C^1x^1 + D^1u^1 \end{aligned} \tag{9}$$

$$s_j: \quad \begin{aligned} \dot{x}^j &= A^jx^j + B^ju^j \\ y^j &= C^jx^j + D^ju^j \end{aligned} \tag{10}$$

$$u^j = h_{j-1,j}y^{j-1} \quad \text{for} \quad j = 2,3,\ldots,m.$$

Let $G_j(\lambda)$ denote the transfer function matrix for s_j, Λ^j the set of distinct eigenvalues of s_j, $b^j_{\ell i}$ the row in B^j related to the last row of the Jordan block for $\lambda_i \in \Lambda^j$. (There will be more than one $b^j_{\ell i}$ when there is more than one Jordan block for $\lambda_i \in \Lambda^j$.)

Theorem 4: A necessary and sufficient condition for controllability of a subsystem s_j in a chain where $\Lambda^k \cap \Lambda^r = \phi$ for $k \neq r$ $k,r \in \{1,2,\ldots,j\}$ is that for each $\lambda_i \in \Lambda^j$, the rows

$$b^j_{\ell i} \prod_{k=2}^{j} h_{k-1,k}G_{k-1}(\lambda_i) \tag{11}$$

are linearly independent.

The proof, which depends on a theorem by Kalman [7], is an extension of a method due to Panda [8]. For details see [4]. Note that $\prod_{k=2}^{j} h_{k-1,k}G_{k-1}(\lambda)$ is the overall transfer function matrix from input to s_j. For the case with local inputs, define $G_{kj}(\lambda)$ to be the transfer function matrix from the local input of subsystem k to the related interaction input of subsystem j in the chain (j > k by assumption). We obtain [4],

Theorem 5: A LSC system, composed of a chain of m controllable subsystems with $\Lambda^j \cap \Lambda^k = \phi$ $j \neq k$ $j,k = 1\ldots m$ and $\Lambda^j_{tt} \cap \Lambda^j_{rr} = \phi$ $t \neq r$ $t,r = 1,2,3$ is controllable if and only if for $j = 2,\ldots,m$, and for each $\lambda_i \in \Lambda^j_{22}$ the rows

$$b^j_{2\ell i}[G_{j-1,j}(\lambda_i) \vdots G_{j-2,j}(\lambda_i) \vdots \cdots \vdots G_{1,j}(\lambda_i)] \qquad (12)$$

are linearly independent.

Theorem 5 leads to various interesting observations. Note that we need only A_{22} in Jordan form in each subsystem and not the whole A matrix. Another point is that given the necessary number of inputs, any subsystems local inputs may be sufficient for controllability of a subsystem. Or a number of "higher level" subsystems may cooperate in controlling one lower in the hierarchy. It has to be pointed out here that the concept of controllability in LSC systems plays a more important role than usual, in that it is also concerned with the effect of the control action through the interconnections. By analyzing controllability insight into local pole placement, stabilizability etc. is gained.

V. STABILIZABILITY AND POLE PLACEMENT IN CHAINS

We shall now apply our remarks related to (5) and Theorem 5 to the stabilizability and pole placement properties of the chain structure.

Theorem 6: A LSC system, composed of a chain of m controllable subsystems with $\Lambda^j \cap \Lambda^k = \phi \quad j \neq k \quad j,k = 1 \ldots m$, and $\Lambda^j_{tt} \cap \Lambda^j_{rr} = \phi \quad t \neq r \quad t,r = 1,2,3$ is locally stabilizable with linear state feedback if and only if $\bigcup_{j=2}^{m} \Lambda^j_{22} \subset C^-$, where C^- is the left-half complex plane.

Proof: Apply the local feedback

$u^j_1 = k^j_1 x^j_1 + k^j_2 x^j_2 + k^j_3 x^j_3$ to s_j, $(j = 2, \ldots, m)$ and

$u^1 = k^1 x^1$ to s_1.

Denoting the new A matrix for the composite system as $\tilde{A}$, we get, after some manipulation,

$$\det(\lambda I - \tilde{A}) = \det(\lambda I - A^1 - B^1 K^1) \prod_{j=2}^{m} \det(\lambda I - A^j_{22}) \cdot \det \begin{bmatrix} \lambda I - A^j_{11} - B^j_{11} K^j_1 & -B^j_{11} K^j_3 \\ -A^j_{21} - B^j_{32} K^j_1 & \lambda I - A^j_{33} - B^j_{33} K^j_3 \end{bmatrix} \qquad (13)$$

from which two facts are seen: a) eigenvalues of the A^j_{22} have not been affected, b) the local feedback of the x^j_2 have been of no use.

The following questions concerning the chain structure can be raised:

(1) If the system cannot be locally stabilized, then which states should be fed back to which subsystems?

(2) At what location in the hierarchy must feedback be inserted to shift a specified pole that cannot be shifted by local feedback?

The answer to the first is "those states of the Jordan form representation related to the mode in question." For an answer to the second question consider Theorem 5 and Equation (12). Check linear independence of $b^j_{2\ell i} G_{j-1,j}(\lambda_i)$. If they are not linearly independent go to $b^j_{2\ell i}[G_{j-1,j}(\lambda_j) \vdots G_{j-2,j}(\lambda_i)]$ then $b^j_{2\ell i}[G_{j-1}(\lambda_i) \vdots G_{j-2,j}(\lambda_i) \vdots G_{j-2,j}(\lambda_i)]$ etc. until a linearly independent set is found. Other combinations can be tested to see if a subset of these systems are also sufficient. If a single subsystem is required through which feedback is to be made, a certain independent set $b^j_{2\ell i} G_{k,j}(\lambda_i)$ has to be found. Note that some of the eigenvalues $\bigcup_{p=k}^{j} \Lambda^P_{22}$ may also be changed in the process.

VI. CONCLUSIONS

The LSC systems structure has been proposed as a model for large scale systems. Certain properties of this structure have been considered. The controllability decomposition canonic form and an algorithm to get a system into this form were given. The canonic form was shown to be very useful in drawing conclusions about controllability, pole placement and stabilizability of LSC systems. Further, it was shown that necessary conditions for controllability of LSC systems are: For any subsystem, for at least one set of necessary predecessors, the Predecessor Composite System has to be output controllable. If the local inputs of the PCS are not adequate, the sequence of steps backwards along preceding PCS's should end at local inputs.

REFERENCES

[1] M. Aoki, "On the Feedback Stabilizability of Decentralized Dynamic Systems," Automatica, March 1972, pp. 163-173.

[2] M. Aoki and M. T. Li, "Controllability and Stabilizability of Decentralized Dynamic Systems," 1973 JACC preprints, pp. 278-286.

[3] S. H. Wang and E. J. Davison, "On the Stabilization of Decentralized Control Systems," IEEE Trans. Automatic Control, Vol. AC-18, October 1973, pp. 473-478.

[4] U. Ozguner and W. R. Perkins, "Controllability, Pole Placement and Stabilizability in Large Scale Composite Systems," Coordinated Science Laboratory, University of Illinois, Urbana, Ill., Rept. R-648, May 1974.

[5] U. Ozguner and W. R. Perkins, "Identification of Necessary Inputs of Linear Time-Invariant Systems," submitted for publication.

[6] U. Ozguner and W. R. Perkins, "On the Multilevel Structure of Large Scale Composite Systems," submitted for publication.

[7] R. E. Kalman, "Irreducible Realizations and the Degree of a Matrix of Rational Functions," SIAM J. Control, Vol. 13, 1965, pp. 520-544.

[8] S. P. Panda, "Controllability of Discrete Composite Systems," Automatica, Vol. 6, 1970, pp. 309-313.

ON EXACT MODEL MATCHING

R. RUTMAN and Y. SHAMASH
School of Engineering
Tel-Aviv University
Tel-Aviv, Israel

ABSTRACT

The problem of "exact model matching" for finite dimensional, linear time-invariant systems is considered. A new algorithm for the solution of the problem is introduced. The approach used is based on the transfer function description of systems. The algorithm is compared with an existing algorithm.

1. INTRODUCTION

An outline of the problem of "exact model matching" for finite dimensional linear time-invariant systems, is given in [1,2]. The problem is a question of finding a state-feedback law for a given system in order that the overall system transfer function becomes exactly equal to a given transfer function.

Basic problems of multivariable control have been treated in the framework of state-variable description of systems. Nevertheless, recent results show that the transfer function concepts proved to be very suitable for certain problems, one of which is, the authors believe, the treated problem.

2. STATEMENT OF THE PROBLEM

Consider the system (A, B, C, D)

$$\begin{aligned} \dot{\underline{x}} &= A\underline{x} + B\underline{u} \\ \underline{y} &= C\underline{x} + D\underline{u} \end{aligned} \qquad (1)$$

where $\underline{x} \in R^n$, $\underline{y} \in R^\ell$ and $\underline{u} \in R^\ell$; A, B, C and D are matrices of compatible dimensions. The linear state-variable feedback (ℓ.s.v.f) control law is

$$\underline{u} = F\underline{x} + K\underline{w} \qquad (2)$$

where $\underline{w} \in R^\ell$; F and K are matrices of compatible dimensions, with K assumed nonsingular to ensure the linear independence of the ℓ external inputs. Substituting for $\underline{u}$ in equation (1) results in

$$\dot{\underline{x}} = (A + BF)\underline{x} + BK\underline{w}, \qquad \underline{y} = (C + DF)\underline{x} + DK\underline{w}. \tag{3}$$

The problem may be stated as follows: find a feedback pair $\{F,K\}$, such that the transfer function matrix of the closed loop system, namely $(C + DF)(sI - A - BF)^{-1}BK + DK$, is identical to the given transfer function matrix of a model system.

The closed loop system of equation (3) may be represented in block diagram form as shown in Fig. 1.

3. THE ALGORITHM

The transfer function matrix of the system (A, B, C, D)

$$G_p(s) = C(sI - A)^{-1}B + D \tag{4}$$

may be written in the form

$$G_p(s) = \frac{P(s)}{q(s)} \tag{5}$$

where

$$q(s) = \det(sI - A), \tag{6}$$

$$P(s) = C.\mathrm{adj}(sI - A).B. \tag{7}$$

Then

$$\underline{y}(s) = G_p(s)\ \underline{u}(s). \tag{8}$$

The affect of the feedback pair $\{F,K\}$ may be produced by the pair $\{H(s),K\}$, as shown in Fig. 2. By doing so we replace equation (2) by the equation

$$\underline{u}(s) = -H(s)\underline{y}(s) + Kw(s) \tag{9}$$

wo that the overall matrix transfer function of the system, should be equal to G(s).

Comparing equations (2) and (9), we have

$$-H(s)\underline{y}(s) = F\underline{x}(s), \tag{10}$$

$$-H(s)[C(sI - A)^{-1}B + D]\underline{u}(s) = F(sI - A)^{-1}B\ \underline{u}(s) \tag{11}$$

and, assuming invertibility of the closed-loop system, we have

$$-H(s)[C(sI - A)^{-1}B + D] = F(sI - A)^{-1}B \tag{12}$$

and, equation (12) leads to,

$$H(s) = -F(sI - A)^{-1}B[C(sI - A)^{-1}B + D]^{-1}$$

$$= -F \frac{\mathrm{adj}(sI - A)B}{\det(sI - A)} \cdot q(s).P^{-1}(s) \tag{13}$$

$$= J(s)P^{-1}(s) \tag{14}$$

where

$$J(s) = -F\ \mathrm{adj}(sI - A).B \tag{15}$$

and $J(s)$ is an $(\ell \times \ell)$ polynomial matrix.

Thus the problem of "exact model matching" by ℓ.s.v.f. has been reduced to computing the pair $\{J(s),K\}$ such that the overall closed loop transfer function is equal to $G(s)$.

It can be seen from Fig. 2, that the closed loop system transfer function may be derived as follows:

$$\underline{u}(s) = K\underline{w}(s) - J(s)P^{-1}(s) \cdot \frac{P(s)}{q(s)} \cdot \underline{u}(s) \tag{16}$$

$$\underline{y}(s) = \frac{P(s)}{q(s)} \cdot \underline{u}(s). \tag{17}$$

Hence

$$\underline{y}(s) = \frac{P(s)}{q(s)} [I + \frac{J(s)}{q(s)}]^{-1} K\underline{w}(s) \tag{18}$$

where I is the identity matrix.

The desired matrix transfer function is of the form

$$G(s) = \frac{U(s)}{v(s)} \tag{19}$$

where $U(s)$ is an $(\ell \times \ell)$ polynomial matrix and $v(s)$ is a scalar polynomial. Comparing (18) and (19), we obtain

$$U^{-1}(s)P(s)v(s) = K^{-1}[q(s)I + J(s)]. \tag{20}$$

Let

$\rho[\cdot] \triangleq$ The degree of the polynomial matrix $[\cdot]$, i.e. the degree of the highest term in s in the matrix polynomial.

Then from equation (15) we have

$$\rho[J(s)] \leq \rho[q(s)] - 1 = n - 1. \tag{21}$$

We also have

$$\rho[P(s)] \leq \rho[q(s)] = n, \tag{22}$$

$$\rho[U(s)] \leq \rho[v(s)]. \tag{23}$$

Hence from equation (20) it can be seen that $K^{-1}[q(s)I + J(s)]$ is a polynomial matrix of degree n. Thus we have the condition that the system can be exactly matched to the desired transfer function only if,

$$U^{-1}(s)P(s)v(s) = \sum_{i=0}^{n} T_i s^i = T(s) \tag{24}$$

where T_i, $i = 0,1,\ldots,n$, are constant matrices and $T_n \neq 0$.

Let

$$q(s) = \sum_{i=0}^{n} q_i s^i \quad , \; q_n = 1. \tag{25}$$

Substituting equations (24) and (25) into equation (20) leads to the following equation,

$$K^{-1} = T_n.$$

Therefore, if T_n^{-1} exists,

$$K = T_n^{-1} \tag{26}$$

$$J(s) = KU^{-1}(s)P(s)v(s) - q(s)I. \tag{27}$$

Thus the conditions that must be satisfied for G(s) to be realized by ℓ.s.f.v., may be summarized as follows:

(a) $U^{-1}(s)P(s)v(s)$ must reduce to a polynomial matrix of degree n.

(b) T_n must be non-singular.

(c) There must exist a unique matrix F such that equation (14) is satisfied.

Deriving F from (15) results in an overdetermined set of $n\ell^2$ equations for $n\ell$ unknowns. This set of equations has a unique solution if, and only if, the solution of $n\ell$ equations produced by any column of the matrix $\operatorname{adj}(sI - A)B$ satisfies the remaining equations.

Because of the structure of equation (15) the solution for the $n\ell$ unknowns reduces to the solution of ℓ sets of n unknowns.

4. COMMENTS AND DISCUSSION

(a) The algorithm introduced in this paper may be summarized as follows:

- (i) Compute $G_p(s)$ using equation (4).
- (ii) Compute T(s), which should reduce to a polynomial matrix. Test if T_n is non-singular.
- (iii) Compute K, equation (26); J(s), equation (27).
- (iv) Compute F, equation (15); and test for consistency.

The computational simplicity of the algorithm can be best appreciated when compared with other algorithms derived for the same problem [1,2].

For example, Wolovich's algorithm includes the transformation of (A, B, C, D) to canonical companion form and the factorization of a rational matrix transfer function into the form $G(s) = X(s)Y^{-1}(s)$, where X(s) and Y(s) are relatively prime polynomial matrices. These operations are computationally cumbersome.

(b) Conceptually the algorithm of this paper is simple and based on the use of equivalent feedback H(s). This naturally leads to the generalization of the problem of "exact model matching" via dynamic feedback.

In future communications the authors hope to treat the generalized problem by using the concept of polynomial equations.

5. EXAMPLE [1]

Given the system (A, B, C, D)

$$A = \begin{bmatrix} -4 & -1 & -3 \\ 3 & 1 & 1 \\ 5 & 1 & 3 \end{bmatrix}; \quad B = \begin{bmatrix} 1 & 1 \\ -1 & 0 \\ -1 & -1 \end{bmatrix}; \quad C = \begin{bmatrix} 4 & -1 & 4 \\ 0 & 0 & 0 \end{bmatrix}; \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}.$$

Find a feedback pair {F,K} such that the desired closed-loop transfer function is

$$G(s) = \frac{U(s)}{v(s)} = \frac{1}{s+3}\begin{bmatrix} 1 & 0 \\ -s & s+3 \end{bmatrix}.$$

From equation (4) we get

$$G_p(s) = \frac{P(s)}{q(s)} = \frac{1}{s^3+4s-2}\begin{bmatrix} s^2+3s+2 & 2s-5 \\ 0 & s^3+4s-2 \end{bmatrix}.$$

Then

$$U^{-1}(s)P(s)v(s) = \begin{bmatrix} s^3+6s^2+11s+6 & 2s^2+s-15 \\ s^3+3s^2+2s & s^3+2s^2-s-2 \end{bmatrix} = \sum_{i=0}^{3} T_i s^i$$

where

$$T_3 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \qquad \Rightarrow \qquad K = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}.$$

Hence

$$J(s) = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} s^3+6s^2+11s+6 & 2s^2+s-15 \\ s^3+3s^2+2s & s^3+2s^2-s-2 \end{bmatrix} - \begin{bmatrix} s^3+4s-2 & 0 \\ 0 & s^3+4s-2 \end{bmatrix}$$

$$= \begin{bmatrix} 6s^2+7s+8 & 2s^2+s-15 \\ -3s^2-9s-6 & -6s+15 \end{bmatrix}$$

F may be computed from equation (15):

$$-\begin{bmatrix} 6s^2+7s+8 \\ -3s^2+9s-6 \end{bmatrix} = \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \end{bmatrix} \begin{bmatrix} s^2 & s^2-s \\ -s^2+s-2 & 2s+1 \\ -s^2+s & -s^2+2s-1 \end{bmatrix}.$$

Using the first column of the right hand polynomial matrix and equating coefficients leads to

$$F = \begin{bmatrix} -13 & 4 & -11 \\ 12 & -3 & 12 \end{bmatrix}.$$

Substituting F back into the above equations shows the solution to be consistent.

6. CONCLUSIONS

The paper reinforces the recently increased belief that the transfer function's concept, enriched by results obtained in the state-variable approach, is viable. In the problem treated here, it resulted in an algorithm which is simple and computationally straightforward. The algorithm does not require a separate test for determining whether the problem of "exact model matching" can be solved via ℓ.s.v.f. The formulation used can be naturally generalized for similar problems involving dynamic compensation.

7. REFERENCES

1. Wolovich, W. A., "The Use of State Feedback for Exact Model Matching," SIAM J. Control, Vol. 10, No. 3, pp. 512-523, 1972.

2. Wang, S. H. and Desoer, C. A., "The Exact Model Matching of Linear Multivariable Systems," IEEE Trans. Aut. Control, Vol. AC-17, No. 3, pp. 347-349, 1972.

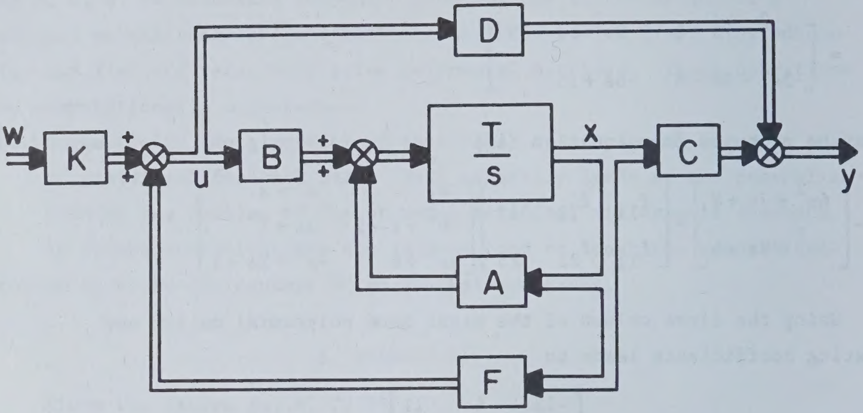

w
K
+
+
u
B
+
+
T
s
x
C
y
D
A
F

FIG. 1

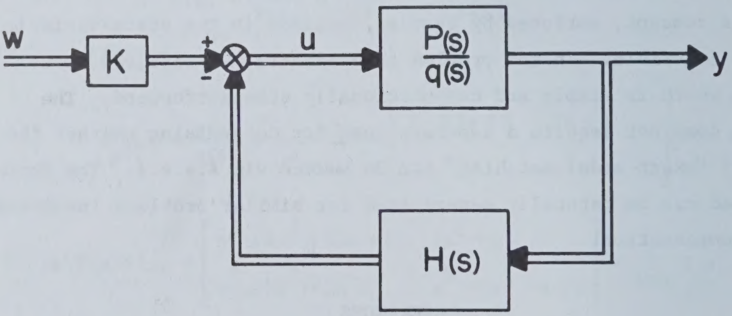

w
K
+
-
u
P(s)
q(s)
y
H(s)

FIG. 2

THE NECESSARY STRUCTURE OF MULTIVARIABLE REGULATORS*

B. A. FRANCIS and W. M. WONHAM
Department of Electrical Engineering
University of Toronto
Toronto, Canada M5S 1A4

The linear time-invariant multivariable system

$$\dot{x}_1 = A_1x_1 + A_3x_2 + B_1u$$
$$\dot{x}_2 = A_2x_2$$
$$y = C_1x_1 + C_2x_2$$
$$z = D_1x_1 + D_2x_2$$

is to be controlled by the dynamic compensator

$$\dot{x}_c = A_cx_c + B_cy$$
$$u = F_cx_c + Fy$$

Here x_1 is the state of the plant, x_2 a vector of disturbances and/or reference signals, u the control, y the vector of measured variables, z the variable to be regulated, and x_c the state of the compensator. The controller is to provide output regulation $(z(t) \to 0$ as $t \to \infty$ for all initial conditions $x_1(0), x_c(0), x_2(0))$ and internal stability $(x_1(t) \to 0$ and $x_c(t) \to 0$ as $t \to \infty$ for all $x_1(0), x_c(0)$, with $x_2(t) \equiv 0)$ in the presence of small parameter variations in certain parts of the system data.

For such structurally stable regulation, the control structure must have the following three features.

---

* This research was partially supported by The National Research Council of Canada, Grant No. A-7399.

(i) Feedback is necessary. By this is meant that y can be written in the form $\begin{pmatrix} w \\ z \end{pmatrix}$ for some vector w, so that the compensator is directly processing the regulated variable z.

(ii) The controller must incorporate an internal model of the disturbance process. Precisely, this means that the Jordan structure of A_c must contain a sufficiently rich version of the Jordan structure of A_2.

(iii) The signal flow topology of the feedback loop must be such that z drives the internal model.

The necessity of these three features we call The Internal Model Principle. A precise mathematical statement and proof are given in the following two papers.

[1] B.A. Francis and W.M. Wonham, The Internal Model Principle for Linear Multivariable Regulators, Appl. Maths. and Optimization, 1 (3), 1974.

[2] B.A. Francis and W.M. Wonham, The Signal Flow Topology of Structurally Stable Linear Regulators, Control Systems Report No. 7419, Dept. of Electrical Engineering, Univ. of Toronto, 1974.

STABILITY CRITERIA FOR TWO-VARIABLE POLYNOMIALS

D. D. Šiljak
School of Engineering
University of Santa Clara
Santa Clara, California

ABSTRACT

Recursive algebraic algorithms are developed for testing various stability properties of two-variable polynomials in finite number of steps. Stability is tested with respect to either the half plane or the unit circle applying only two times the equivalent of the Routh or Marden test regardless of the degree of the polynomial.

I. INTRODUCTION

In classical stability theory of linear systems [1], the problem is to determine the location of zeros of a real or complex polynomial relative to the half plane or the unit circle. A large number of solutions to the stability problem as a set of algebraic inequalities involving only polynomial coefficients are available as variations of the Hermite and Routh-Hurwitz criteria for the half plane, and Schur-Cohn and Marden-Jury criteria for the unit circle.

Theory of electric networks brought forward as a fundamental concept the notion of positive real functions and matrices [1]. The positive real property has two essential aspects, one being stability of a real polynomial with respect to the open half plane or the unit disc, the other being nonnegativity of a real polynomial on the imaginary axis or the unit circle. It was shown only recently [2, 3] how nonnegativity of polynomials and polynomial matrices can be resolved in the context of the algebraic stability algorithms, and test the positive realness of real rational functions and matrices by a single recursive scheme in finite number of steps.

When the positive real concept in network theory was extended to multivariable functions by Ożaki and Kasami [4], the stability problem of multivariable polynomials arose in a natural way. For two-variable polynomials, the stability problem was reduced by Ansell [5] to testing stability of a real polynomial in the open half plane and nonnegativity of a set of real polynomials on the imaginary axis. The set of polynomials was generated by the principal minors of the corresponding Hermite matrix, and their number was equal to the order of the matrix. Subsequently, the stability problem of two-dimensional digital filters considered by Shanks, et. al. [6], was resolved by Huang [7] using the solution mechanism of Ansell [5] and the bilinear transformation. Once the direct methods [3] were available for testing nonnegativity of polynomials and polynomial matrices on the unit circle, it was possible to avoid bilinear transformation in this context and use the Schur-Cohn criterion [8]. As an alternative to the Schur-Cohn criterion, the Marden-Jury table can be used as shown in [9]. In [9], however, the Sturm's test was recommended for testing positivity of scalar polynomials on the finite interval, which is inferior to the direct methods [3] both conceptually and algorithmically.

The purpose of this paper is to combine the classical stability conditions for one-variable polynomials [1] with algebraic criteria for positiv-

---

*The research reported herein was supported by NASA Grant NGR 05-017-010.

ity of polynomial matrices [2, 3] developed in the framework of positive reality, and produce direct recursive algorithms for testing stability properties of two-variable polynomials. The outstanding simplicity of the algorithms is achieved by utilizing the fact [3, 10] that positivity of a polynomial matrix is equivalent to positivity of the matrix at a single value of the independent variable and positivity of its determinant. Stability of two-variable polynomials is tested with respect to either the open half plane or the unit disc by executing only two times the equivalent of the Routh or Marden tests regardless of the degree of the polynomials involved. The proposed algorithms are completely recursive and are performed by finite number of algebraic calculations, which makes them suitable for computer implementation [11].

II. POSITIVITY OF MATRIX POLYNOMIALS

Let us consider a regular, real matrix polynomial*

$$F(z) \equiv \sum_{k=0}^{N} A_k z^k , \tag{1}$$

where A_k are real, constant $m \times m$ coefficient matrices, $A_N \neq 0$, and $z = \sigma + i\omega$ is the complex variable. By following [2, 3], we will outline the necessary and sufficient conditions for the Hermitian part

$$H(z) = \frac{1}{2}\,[F(z) + F^*(z)] \tag{2}$$

of $F(z)$ to be positive either on the imaginary axis $\mathcal{I} = \{z\colon \operatorname{Re} z = 0\}$ - axis positivity; or positive on the unit circle $\mathcal{C} = \{z\colon |z| = 1\}$ - circle positivity. That is, for all constant complex vectors $U \neq 0$ of a unitary n-dimensional vector space, we have either

$$U^* H(z)\, U > 0 , \quad \forall z \in \mathcal{I} \tag{3}$$

or

$$U^* H(z)\, U > 0 , \quad \forall z \in \mathcal{C} . \tag{4}$$

Let us first consider the *axis positivity* of $H(z)$. We replace z by $i\omega$ in $H(z)$ and use $f(\omega^2) \equiv \det H(i\omega)$, which is a real even scalar polynomial

$$f(\omega^2) \equiv \sum_{k=0}^{n} a_{2k}\omega^{2k} , \tag{5}$$

where $a_{2n} \neq 0$. By forming the modified Routh array [2] corresponding to $f(\omega^2)$,

$$\begin{array}{llllll}
r_0 = (-1)^n a_{2n} & (-1)^{n-1} a_{2n-2} & \cdot\ \cdot\ \cdot\ \cdot & -a_2 & a_0 \\
r_1 = (-1)^n n a_{2n} & (-1)^{n-1}(n-1) a_{2n-2} & \cdot\ \cdot\ \cdot & -a_2 & \\
\quad\vdots & \quad\vdots & & & \\
r_{2n} = \quad a_0 & & & &
\end{array} \tag{6}$$

---

*With some obvious exceptions, capital Roman letters denote matrices and vectors, lower case Roman and Greek letters denote scalars, and capital italic letters denote sets.

we conclude that $f(\omega^2) \not\equiv a_0$ is positive for all ω,

$$f(\omega^2) > 0 \ , \ \forall\omega \tag{7}$$

if and only if $v(f) = v[r_0, r_1, \ldots , r_{2n}] = n$ and $a_0 > 0$, where $v(f)$ is the number of sign changes among the Routh numbers $r_0, r_1, \ldots , r_{2n}$; or $f(\omega^2) \equiv a_0 > 0$. [2]

We establish the following:

Theorem 1. A Hermitian polynomial $m \times m$ *matrix* $H(z)$ *is axis positive if and only if the polynomial* $f(\omega^2) \equiv \det H(i\omega)$ *satisfies*

$$v(f) = n \ , \ a_0 > 0 \tag{8}$$

and $H(0) > 0$.

Proof. Since (8) is equivalent to $\det H(i\omega) > 0 \ , \ \forall\omega$, it remains to show that [10]

$$H(z) > 0 \ , \ \forall z \in I \tag{9}$$

implies and is implied by

$$\begin{aligned} &\text{(i)} \quad \det H(i\omega) > 0 \ , \ \forall\omega \\ &\text{(ii)} \qquad H(0) > 0 \ . \end{aligned} \tag{10}$$

Obviously, (9) implies (10). Let (10) be satisfied. Then, from (10i), $\det H(i\omega) \neq 0 \ , \ \forall\omega$; that is, the eigenvalues $\lambda_k(\omega)$, $\forall k = 1, 2, \ldots , m$, of $H(i\omega)$ are different from zero for all ω. Since $\lambda_k(\omega)$ are real and continuous functions of ω, and for $\omega = 0$ they are all positive due to (10ii), they are positive for all ω. Consequently, (10) implies (9) and the proof of Theorem 1 is complete.

Let us now consider the *circle positivity* of $H(z)$. The polynomial $\tilde{f}(z) \equiv \det H(z)$ is a real self-inversive scalar polynomial

$$\tilde{f}(z) = \sum_{k=0}^{n} a_k (z^k + z^{-k}) \tag{11}$$

where $a_n \neq 0$.

Similarly, as in Theorem 1, we need first to determine positivity of $\tilde{f}(z)$ on C. We recall [3] that

$$\tilde{f}(z) > 0 \ , \ \forall z \in C \tag{12}$$

if and only if the self-inversive polynomial

$$f(z) \equiv \sum_{k=0}^{n} a_k (z^{n+k} + z^{n-k}) \tag{13}$$

defined by $f(z) = z^n \tilde{f}(z)$ is such that either $f(z) \not\equiv a_0$ has no zeros on the unit circle C and $\sum_{k=0}^{n} a_k > 0$; or $f(z) \equiv a_0 > 0$.

To eliminate the possibility of zeros on C of the polynomial $f(z)$ we can use the Marden's recursive scheme [12] as proposed in [3]. With an arbitrary polynomial $f(z) \equiv \sum_{k=0}^{n} a_k z^k$, $a_n \neq 0$, we associate the polynomial

$$f^{\square}(z) = z^n f(z^{-1}) = \sum_{k=0}^{n} a_k z^{n-k} , \tag{14}$$

and construct the sequence of polynomials

$$f_j(z) = \sum_{k=0}^{n-j} a_k^{(j)} z^k \tag{15}$$

where $f_0(z) \equiv f(z)$, and

$$f_{j+1}(z) = a_0^{(j)} f_j(z) - a_{n-j} f_j^{\square}(z) , \quad j = 0, 1, \ldots , n-1 , \tag{16}$$

so that

$$a_k^{(j+1)} = a_0^{(j)} \; a_k^{(j)} - a_{n-j}^{(j)} \; a_{n-j-k}^{(j)} . \tag{17}$$

We denote the constant term $a_0^{(j)}$ of each polynomial $f_j(z)$ by δ_j ,

$$\delta_{j+1} = [a_0^{(j)}]^2 - [a_{n-j}^{(j)}]^2 = a_0^{(j+1)} , \quad j = 0, 1, \ldots , n-1 , \tag{18}$$

and form the products

$$\Pi_k(f) = \delta_1 \delta_2 \ldots \delta_k , \quad k = 1, 2, \ldots , n . \tag{19}$$

Then, as proved by Marden [12], if $\nu = \nu(f)$ of the products $\Pi_k(f)$ negative and the remaining $\pi = n - \nu$ are positive, then $f(z)$ has no zeros on the unit circle $C = \{z: |z| = 1\}$, ν zeros inside C , and π zeros outside C . As shown by Jury [13], the simple necessary conditions for the case $\nu = n$ are $f(1) > 0$ and $(-1)^n f(-1) > 0$, or

$$\sum_{k=0}^{n} a_j > 0 , \quad (-1)^n \sum_{k=0}^{n} (-1)^k a_k > 0 . \tag{20}$$

When the polynomial to be considered is a self-inversive, as is the case with $f(z)$ in (11), then $f^{\square}(z) \equiv f(z)$ and $f_1(z) \equiv 0$. Thus, we have a singular polynomial sequence [12]. On the basis of results obtained in [3], we derive a finite recursive algorithm for counting the number of zeros of $f(z)$ on the unit circle C .

Let us construct the polynomial sequence for the polynomial $f(z)$, and then enumerate the polynomials that precede identically zero polynomials in the sequence by f_{j_μ} , $(\mu = 1, 2, \ldots , \zeta)$. To continue the sequence, each identically zero polynomial f_{j_μ} is replaced by the newly formed polynomial $[f'_{j_\mu}(z)]^{\square} = z^{2n-(j_\mu+1)} f'_{j_\mu}(z^{-1})$ which has the same number of zeros inside the unit circle as proved by Cohn [14], and where $f'_{j_\mu}(z^{-1})$ is obtained by differentiating $f_{j_\mu}(z)$ with respect to z and then substituting z^{-1} for z . Let us denote by ν_μ the number of the negative products Π_k which correspond to the polynomials $f_{j_\mu+1}$, $f_{j_\mu+2}$, ... , $f_{j_{\mu+1}}$. That is, ν_μ

is the number of negative products Π_k between the two consecutive zero rows j_μ and $j_{\mu+1}$. By going through the matter chronologically (similarly as in [2]), we conclude that the number $\nu(f)$ of zeros inside C of the polynomial $f(z)$ is

$$\nu(f) = \sum_{i=1}^{\zeta} \nu_i , \tag{21}$$

and the number $\rho(f)$ of zeros on the circle C is

$$\rho(f) = 2[n - \nu(f)] . \tag{22}$$

Now, as in [2], we arrive at the following:

Theorem 2. *A Hermitian polynomial* $m \times m$ *matrix* $H(z)$ *is circle positive if and only if the polynomial* $f(z) \equiv z^n \det H(z)$ *satisfies*

$$\nu(f) = n , \sum_{k=0}^{n} a_k > 0 \tag{23}$$

and $H(1) > 0$.

Proof. In order to prove Theorem 2, we notice first that (23) is equivalent to $\det H(z) > 0, \forall z \in C$. Then, it remains to show that

$$H(z) > 0 , \forall z \in C \tag{24}$$

implies and is implied by

$$\text{(i)} \quad \det H(z) > 0 , \forall z \in C$$

$$\text{(ii)} \quad H(1) > 0 . \tag{25}$$

Using the same argument as in the proof of Theorem 1, the equivalence between (24) and (25) is easily established. This proves Theorem 2.

Axis positivity conditions of Theorem 1 were used to test absolute stability of multi-nonlinear systems in [15]. Axis nonnegativity of matrix polynomials can also be verified by the modified Routh test, but the conditions require more computations [2]. Such extended tests can be used effectively to verify optimality and passivity of linear multivariable systems [15]. Similarly, the conditions of Theorem 2 can be used to determine absolute stability properties of discrete nonlinear systems, and can also be extended to provide conditions for circle nonnegativity of polynomial matrices as shown in [3].

To give an estimate of the efficiency of the direct positivity tests when programmed for computer applications, Schaldach [11] calculated the execution time to be approximately equal to $0.025n^2 + 1$ seconds where n is the degree of the tested polynomial. When standard root-solving programs were used to conclude positivity (e.g. IBM Scientific Subroutine: 1130-CM-02X) the execution time was ten to a hundred times that of the direct test. Needless to say that the root-solving schemes are iterative and cannot be performed in finite number of steps. Furthermore, they are highly sensitive to multiplicity of the zeros of the tested polynomials, the feature that has no effect on the direct tests. Because of these apparent advantages, the direct positivity tests were used in [11] to deliniate regions of positivity with respect to two adjustable coefficients of polynomials up to the tenth degree. Typical computation time was 15 minutes to 1 hour. During this time, some 3000 positivity tests were performed, a task that is virtually impossible to complete by using root-solving schemes or the Sturm's test.

In the following section, we will show how the direct positivity tests can be effectively applied to test numerically various stability properties of two-variable polynomials.

3. STABILITY

Let us consider a real two variable polynomial

$$h(s, z) \equiv \sum_{p=0}^{\ell} \sum_{q=0}^{m} b_{pq} s^p z^q \tag{26}$$

where s and z are the two complex variables, and for some p, q, the coefficitent b_{p} and b_{mq} are not both zero. We are interested in determining the conditions under which the polynomial $h(s, z)$ is stable, that is, $h(s, z)$ has all its zeros in the left-half $\mathcal{P} = \{s: \text{Re } s < 0\}$ of the s plane and in the unit disc $\mathcal{D} = \{z: |z| < 1\}$ of the z plane. The desired conditions for the stability property

$$h(s, z) \neq 0 , \quad \{s \in \mathcal{P}^c\} \cap \{s \in \bar{\mathcal{D}}\} \tag{27}$$

are obtained in terms of positivity of polynomial Hermitian matrices considered in the previous section.

Following Ansell [5], we can show that (27) is equivalent to

$$\begin{aligned} &\text{(i)} \quad h(s, 0) \neq 0 , \quad s \in \mathcal{P}^c \\ &\text{(ii)} \quad h(i\omega, z) \neq 0 , \quad z \in \bar{\mathcal{D}} \end{aligned} \tag{28}$$

Obviously, (27) implies (28). Let us assume that (28) holds. The implicit relationship $h(s, z) = 0$ defines the algebraic function $z = u(s)$, where we assume that $u(s)$ has no singularities for all $s \in P^c$. If $u(s)$ has singularities, there, they can be removed by the usual detour similarly as in [7]. Now, from (28ii) we have that $|u(i\omega)| > 1$, which by maximum modulus theorem and (28i) implies $|u(s)| > 1$, $\forall s \in P^c$, as shown in [5]. Therefore, (28) implies (27).

All one needs to test the stability property (27) of the two-variable polynomial $h(s, z)$ is the Routh table [2]. It is easy to see that it is true for condition (28i]. We use the original Routh table [16] to conclude that the polynomial $g(s) \equiv h(s, 0)$ has all its zeros in P. To test condition (28ii), we can apply the modified Routh array [2]. In analogy to Ansell's use of the Hermite matrix, one can form first the Schur-Cohn matrix [1, 8] for the polynomial $c(z) \equiv z^m h(i\omega, z^{-1})$

$$c(z) \equiv \sum_{q=0}^{m} c_q z^q \tag{29}$$

where the coefficients c_q are given as

$$c_q = \sum_{p=0}^{\ell} b_{p,n-q} s^p \tag{30}$$

and $s = i\omega$. The $m \times m$ Schur-Cohn matrix $C = (c_{pq})$ has the elements defined by

$$c_{pq} = \sum_{k=1}^{p} (c_{m-p+k}\,\bar{c}_{m-q+k} - \bar{c}_{p-k}\,c_{q-k}), \quad p \le q . \tag{31}$$

In (31), $\bar{c}_i$ is the complex conjugate of c_i .

We recall [1] that $C > 0$ if and only if $c(z) = 0$ implies $z \in \mathcal{D}$. Since $C = C(i\omega)$ defined by (31) is a Hermitian matrix $(C = C^*)$, we have the polynomial $f(\omega^2) \equiv \det C(i\omega)$ as in (5), and can establish $C(s) > 0$, $\forall s \in \mathcal{I}$, by applying the modified Routh algorithm to $f(\omega^2)$ and Sylvester inequalities to $C(0) > 0$. Therefore,

Theorem 3. A real two-variable polynomial $h(s, z)$ *has the stability property* (27) *if and only if the polynomials* $g(s) \equiv h(s, 0)$ *and* $f(\omega^2) \equiv \det C(i\omega)$ *satisfy*

$$\begin{aligned} &\text{(i)} \quad g(s) = 0 \Rightarrow s \in \mathcal{P} \\ &\text{(ii)} \quad v(f) = n \;,\; a_0 > 0 \end{aligned} \tag{32}$$

and $C(0) > 0$.

Proof. Theorem 3 follows automatically from (28) and Theorem 1.

Stability of two-variable polynomials arise naturally in the network analysis when positive reality of functions of two complex variables [4, 5] is used. In such cases, we are interested in establishing necessary and sufficient conditions for the polynomial $h(s, z)$ to have the stability property

$$h(s, z) \neq 0, \quad \{s \in \mathcal{P}^c\} \cap \{z \in \mathcal{P}^c\}. \tag{33}$$

As shown by Ansell [5], the property (33) is equivalent to

$$\begin{aligned} &\text{(i)} \quad h(s, 1) \neq 0 \;, \quad s \in \mathcal{P}^c \\ &\text{(ii)} \quad h(i\omega, z) \neq 0, \quad z \in \mathcal{P}^c . \end{aligned} \tag{34}$$

Again, to test (34i) we can use the Routh test. To verify (34ii) we follow Ansell's approach and consider the polynomial $d(z) \equiv h(i\omega, z)$,

$$d(z) \equiv \sum_{q=0}^{m} d_q z^q , \tag{35}$$

where

$$d_q = \sum_{p=0}^{\ell} b_{pq} s^p , \tag{36}$$

and $s = i\omega$. With $d(z)$ we associate the $m \times m$ Hermite matrix $D = (d_{pq})$ with elements d_{pq} defined by [16]

$$d_{pq} = \begin{cases} (-1)^{\frac{1}{2}(p+q)} \sum\limits_{k=1}^{p} (-1)^k (d_{m-k+1}\,\bar{d}_{m-p+k} - \bar{d}_{m-k+1}\,d_{m-p-q+k}), & p+q \text{ even} \\ (-1)^{\frac{1}{2}(p+q+1)} \sum\limits_{k=1}^{p} (-1)^k (d_{m-k+1}\,\bar{d}_{m-p+k} - \bar{d}_{m-k+1}\,d_{m-p-q+k}), & p+q \text{ odd} \end{cases}$$

and $p \leq q$.

We remember [16] that $D > 0$ if and only if $d(z) = 0$ implies $z \in \mathcal{P}$. Since $D = D(i\omega)$ is a Hermitian matrix $(D = D^*)$, we have $f(\omega^2) \equiv \det D(i\omega)$ as in (5), and we can state the following:

Theorem 4. A real two-variable polynomial $h(s, z)$ *has the stability property* (34) *if and only if the polynomials* $g(s) \equiv h(s, 1)$ *and* $f(\omega^2) \equiv \det D(i\omega)$ *satisfy*

$$\text{(i)} \quad g(s) = 0 \Rightarrow s \in \mathcal{P}$$

$$\text{(ii)} \quad \nu(f) = n \ , \ a_0 > 0 \tag{37}$$

and $D(0) > 0$.

Proof. Theorem 4 follows directly from (34) and Theorem 1.

As in the case of Theorem 3, we need only the Routh algorithm to test stability property (33) of the two-variable polynomial $h(s, z)$. If the result of Theorem 4 is compared to that of Ansell [5], we notice that instead of testing positivity of all leading principal minors of the matrix $D(i\omega)$, we need to test only positivity of the determinant of $D(i\omega)$. Furthermore, the Sturm algorithm proposed by Ansell for testing positivity of minors of $D(i\omega)$, is replaced by the systematic Routh test available in the efficient and familiar table form (6) [2].

In the stability analysis of recursive digital filters [6-8], it is of interest to establish necessary and sufficient conditions for a polynomial $h(s, z)$ to have the stability property

$$h(s,\ z) \neq 0, \quad \{s \in \bar{\mathcal{D}}\} \cap \{z \in \bar{\mathcal{D}}\}. \tag{38}$$

By following Huang [7], one can show that (38) is equivalent to

$$\text{(i)} \quad h(s,\ 0) \neq 0, \quad s \in \bar{\mathcal{D}}$$

$$\text{(ii)} \quad h(e^{i\omega},\ z) \neq 0, \quad z \in \bar{\mathcal{D}} . \tag{39}$$

Condition (39i) can be tested by the classical recursive algorithm of Marden [11] described by equations (14)-(19). That is, the algorithm is used to show that $\nu(g)$ is equal to the degree of the polynomial $g(s) \equiv s^{\ell} h(s^{-1},0)$.

To test condition (39ii) in analogy to Ansell's approach [5], we consider the polynomial $c(z) \equiv z^m h(e^{i\omega}, z^{-1})$, as in (29) with coefficients c_q defined by (30) and $s = e^{i\omega}$. With the polynomial $c(z)$ we associate again the Schur-Cohn $m \times m$ matrix [8] $C = (c_{pq})$ specified by (31). The matrix $C(e^{i\omega})$ is a Hermitian matrix $(C = C^*)$, and we have $f(e^{i\omega}) \equiv \det C(e^{i\omega})$, where $f(z)$ is a self-inversive polynomial as in (13). Then, we arrive at the following:

Theorem 5. A real two-variable polynomial $h(s, z)$ *has the stability property* (39) *if and only if the polynomials* $g(s) \equiv h(s, 0)$ *and* $f(z) \equiv \det C(z)$ *satisfy*

$$\text{(i)} \quad g(s) = s^{\ell} h(s^{-1},0)$$

$$\text{(ii)}\quad \nu(f) = n\ ,\quad \sum_{k=0}^{n} a_k > 0 \tag{40}$$

and $C(1) > 0$.

Proof. Theorem 5 follows directly from (39) and Theorem 2.

The result of Theorem 5 is considerably simpler than that of Anderson and Jury [8], and Maria and Fahmy [9], in that only $\det C(z)$ need to be tested for positivity on $\mathcal{C}$ instead of all the m leading principal minors of the matrix $C(z)$ as proposed in [8], or the same number of the first column entries in the Jury's table as required in [9]. Moreover, positivity of $\det C(z)$ on $\mathcal{C}$ can be carried out by the recursive scheme of Marden [12] which can be efficiently tabulated as shown by Jury [13]. Recently, the Marden scheme was rewritten by Barnett [17] in terms of the Routh algorithm, which made it possible to determine the stability property (39) by only one algorithm of extreme simplicity - that of Routh. This proposition, however, requires that the Barnett's scheme of Routh be extended to incorporate the self-inversive polynomials and singular sequences in an economical way.

4. CONCLUSION

Recursive algorithms were outlined for testing stability of real two-variable polynomials with respect to the open half plane and the unit disc. Half plane properties were tested by the Routh table, and the Marden method was used to determine the stability properties relative to the unit circle. When compared with the existing stability criteria [5-8], the algorithms offer an outstanding simplicity in that arbitrary n-th order polynomials in two variables can be tested by applying only two times the equivalent of the Routh and/or Marden test, and checking positive definitness of the corresponding constant Hermitian n-th order matrix. The simplicity is a consequence of the effective solution to the problem of positivity of polynomial matrices presented in [2, 3, 15] and the suitability of the solution for computer implementation [11].

The outlined algorithms can be extended to multi-variable polynomials [18-20], and are presently used in the development of positive real tests for multi-variable functions and matrices.

REFERENCES

1. Barnett, S., "Matrices in Control Theory", Van Nostrand, London, 1971.

2. Šiljak, D. D., "New Algebraic Criteria for Positive Realness", J. Franklin Inst., 291(1971), 109-120.

3. Šiljak, D. D., "Algebraic Criteria for Positive Realness Relative to the Unit Circle", ibid, 295(1973), 469-476.

4. Ozaki, H., and T. Kasami, "Positive Real Functions of Several Variables and Their Applications to Variable Networks", IRE Trans., CT-7 (1960), 251-260.

5. Ansell, H. G., "On Certain Two-Variable Generalizations of Circuit Theory with Applications to Networks of Transmission Lines and Lumped Reactances" IEEE Trans., CT-11 (1964), 214-223.

6. Shanks, J. L., S. Treitel, and J. H. Justice, "Stability and Synthesis of Two-Dimensional Recursive Filters", ibid., AU-20 (1972), 115-128.

7. Huang, T. S., "Stability of Two-Dimensional Recursive Filters", ibid., AU-20 (1972), 158-163.

8. Anderson, B.D.O., and E. I. Jury, "Stability Test for Two-Dimensional Recursive Filters", ibid., AU-21 (1973), 366-372.

9. Maria, G. A., and M. M. Fahmy, "On the Stability of Two-Dimensional Digital Filters", ibid., AU-21(1973), 470-472.

10. Jury, E. I., and B. W. Lee, "A Stability Theory for Multinonlinear Control Systems, Paper No. 28.a, Third IFAC Cong., London, 1966, pp. 28A.1-11.

11. Schaldach, B. H. D., "A Computer Algorithm to Test for Positive Realness" M.S. Thesis, University of Santa Clara, California, January, 1973.

12. Marden, M., "Geometry of Polynomials", American Mathematical Society, Providence, R. I., 1966.

13. Jury, I. E., "The Theory and Applications of Z-Transform Methods", John Wiley, New York, 1964.

14. Cohn, A., "Über die Anzahl der Wurzeln einer algebraischen Gleichung in einem Kreise", Math. Z., 14 (1922), 110-148.

15. Šiljak, D. D., "A Criterion for Nonnegativity of Polynomial Matrices With Application to System Theory", Publ. Facultě Élec. Univ. Belgrade, Sěrie: Electronique, Telecommunications, Automatique, 96 (1973), 163-172.

16. Lehnigh, S. H., "Stability Theorems for Linear Motions", Prentice-Hall, Englewood Cliffs, N. J., 1966.

17. Barnett, S., "Application of the Routh Array to Stability of Discrete-Time Linear Systems", Int. J. Control, 19 (1947), 47-55.

18. Jury, E. I., "Stability of Multidimensional Discrete and Continuous Filters: Inners Approach", Proc. 1974 IEEE Intl. Symposium Circuits and Systems, San Francisco, California, April 22-24, 1974, pp. 70-75.

19. Jury, E. I., "Inners and Stability of Dynamic Systems", Wiley, New York, 1974 (to be published).

20. Bose, N. K., and E. I. Jury, "Positivity and Stability Tests for Multidimensional Filters (Discrete-Continuous)", IEEE Trans., ASSP-22 (1974), 174-180.

ON THE STABILIZATION OF LINEAR SYSTEMS

MEDHAT M. EL-GENEIDY
Electrical Engineering Department
Alexandria University, Alexandria, Egypt

ROGER L. BERGER
McDonnell Douglas Corporation
St. Louis, Missouri

ABSTRACT

The stabilization of linear continuous systems is studied using an algebraic approach. Two classes of linear continuous systems are investigated: a certain class of time-varying systems, and time-invariant systems with time delays in the state and control. In each case a necessary and sufficient condition for stabilizing the specific class of linear continuous systems is presented. As discussed in Reference (1), the results developed here can be specialized to give a necessary and sufficient condition for stabilizing time-invariant systems.

I. INTRODUCTION

The problem of stabilizing linear time-invariant systems has been studied extensively (see for example the work of Wonham in Reference (2)). However, the problem of stabilizing time-varying systems and stabilizing systems with delays has not received nearly as much attention. In this paper the problem of stabilizing a special class of time-varying systems, namely the class of constant rank systems with bounded parameters introduced by Silverman and Meadows in Reference (3), and the class of time-invariant systems with time delays in the state and control, is studied.

In Reference (4), Wolovich derived a sufficient condition for stabilizing a certain class of controllable time-varying systems by the use of static state feedback. In this paper, a larger class of time-varying systems is considered which includes, as a special case, the class of systems considered by Wolovich in Reference (4). This class of systems need not be completely controllable. A necessary and sufficient condition for stabilizing this class of systems is given. The analysis is made through the concept of uncontrollable modes.

In Reference (5), Ross derived a sufficient condition for the existence of a state variable feedback control for systems having a delay in the state only, such that a certain quadratic performance index has a finite value. This condition implies that the system under consideration is stabilizable. An additional sufficient condition for stabilizing such systems is given in Reference (6). In this paper, a larger class of time invariant systems with time delays is considered which includes, as a special case, the class of systems considered in References (5) and (6). Here the delays are assumed to be in the state and control. A necessary and sufficient condition for stabilizing this class of delayed differential systems is given.

An algebraic approach was utilized to develop the theoretical results summarized in this paper. The required proofs, which are discussed in detail in Reference (1), are not included here because of space limitations.

II. TIME VARYING SYSTEMS

The problem of stabilizing a class of time-varying systems has been studied by Wolovich in Reference (4). It can be shown that the class of systems considered in Reference (4) is, in fact, a class of completely controllable constant rank systems. In Reference (4), Wolovich utilized the reducibility properties of a certain class of linear autonomous systems, discussed in detail in Reference (7), to give a sufficient condition for stabilizing such systems. He showed that for a system in this class it is possible to find a state feedback compensator such that the closed loop system matrix is reducible to a constant matrix whose eigenvalues can be arbitrarily specified. This automatically implies that this time-varying system is stabilizable.

In this section the work of Wolovich is extended. A necessary and sufficient condition for stabilizing constant rank systems with bounded parameters is given. The assumption of complete controllability is relaxed and the systems treated here need not to be completely controllable.

The class of systems to be considered here is given by:

$$\dot{x}(t) = A(t)\, x(t) + B(t)\, u(t) \qquad (1.a)$$

$$y(t) = C(t)\, x(t) \qquad (1.b)$$

where $x(t) \varepsilon R^n$ is the state vector, $u(t) \varepsilon R^r$ is the control vector and $y(t) \varepsilon R^m$ is the output vector. The matrices $A(t)$, $B(t)$ and $C(t)$ are bounded continuous functions of time, sufficiently differentiable and of dimensions compatible with the vectors $x(t)$, $u(t)$ and $y(t)$. It is assumed that $B(t)$ and $C(t)$ are of full rank for all $t \geq t_o$. For the sake of completeness the class of constant rank systems is defined first.

Define (Reference (8)), the following generalized controllability and observability matrices:

$$W_a(t) = [U_1(t) \;\; U_2(t) \; - - - - - \; U_a(t)]$$

$$R_b'(t) = [S_1'(t) \;\; S_2'(t) \; - - - - - \; S_b'(t)]$$

where:

$$U_{k+1}(t) = A(t)\, U_k(t) - \frac{d}{dt}\, (U_k(t)) \qquad U_1(t) = B(t)$$

$$S_{k+1}(t) = S_k(t)\, A(t) + \frac{d}{dt}\, (S_k(t)) \qquad S_1(t) = C(t)$$

By definition, the system (1) is a constant rank system if and only if there exist positive integers a, b, p and q such that $A(t)$ is max(a,b)-1 continuously differentiable, $B(t)$ is "a" times continuously differentiable and $C(t)$ is "b" times continuously differentiable and such that:

$$\text{rank } W_a(t) = \text{rank } W_{a+1}(t) = p \leq n \quad \text{for all } t \varepsilon R$$

$$\text{rank } R_b(t) = \text{rank } R_{b+1}(t) = q \leq n \quad \text{for all } t \varepsilon R$$

It was shown in Reference (8) that the system (1) is completely controllable (completely observable) if and only if p=n (q=n).

Definition II.1

Consider the linear transformation:

$$y(t) = T(t)\ x(t)$$

where $T(t)$ is an nxn matrix valued function. $T(t)$ is said to be a Lyapunov transformation if there exists $t_1 \geq 0$ such that:

(i) $T(t)$ and $\dot{T}(t)$ are continuous and bounded on $[t_1, \infty]$

(ii) $|\det[T(t)]| > 0$ for all $t \varepsilon\ [t_1, \infty]$

Definition II.2

Any nxn matrix valued function $A(t)$ will be termed reducible to a constant matrix Y for all $t\varepsilon\ [t_1, t_2]$ if there exists a Lyapunov transformation $T(t)$ defined on $[t_1, t_2]$ such that, for all $t\varepsilon\ [t_1, t_2]$:

$$[\dot{T}(t) + T(t)\ A(t)]\ T^{-1}(t) = Y$$

Lemma II.1

Consider the system:

$$\dot{x}(t) = A(t)\ x(t)$$

Assume that there exist two Lyapunov transformations $T_1(t)$ and $T_2(t)$ such that:

$$A(t) \xrightarrow{T_1(t)} Y_1$$

$$A(t) \xrightarrow{T_2(t)} Y_2$$

($A(t)$ is reducible to Y_i by the Lyapunov transformation $T_i(t)$.) Then:

$$S(Y_1) = S(Y_2)$$

($S(Y_i)$ is the set of eigenvalues of the square matrix Y_i.)

Lemma II.2

Consider the linear system:

$$\dot{x}(t) = A(t)\ x(t)$$

Let $A(t)$ be reducible to Y_1. Let $T(t)$ be any bounded differentiable linear transformation for all $t\varepsilon\ [t_o, \infty]$ and let:

$$\overline{A}(t) = \dot{T}(t)\ T^{-1}(t) + T(t)\ A(t)\ T^{-1}(t)$$

then $\overline{A}(t)$ is also reducible to Y_1.

Lemma II.3

Let $A(t)$ be in the form:

$$A(t) = \begin{bmatrix} A_{11}(t) & 0 \\ & \\ A_{21}(t) & A_{22}(t) \end{bmatrix}$$

Assume that $A_{11}(t)$ is reducible to Y_1 and $A_{22}(t)$ is reducible to Y_2. Assume also that $e^{Y_2 t} M e^{-Y_1 t} \to 0$ as $t \to \infty$ for some M of appropriate dimensions. If $A_{21}(t)$ is bounded, then A(t) is reducible to Y with:

$$S(Y) = S(Y_1) \cup S(Y_2)$$

Corollary II.1

Let:

$$A(t) = \begin{bmatrix} A_{11}(t) & A_{12}(t) & --- & A_{1n}(t) \\ 0 & A_{22}(t) & --- & A_{2n}(t) \\ - & - & & - \\ - & - & & - \\ 0 & 0 & --- & A_{nn}(t) \end{bmatrix}$$

Assume that $A_{ii}(t)$ is reducible to the constant matrix Y_i for each i from 1 to n. Assume also that $e^{Y_1 t} M e^{-Y_i t} \to 0$ as $t \to \infty$ for some M, i = 2,3, ---, n. If A(t) is bounded then it is reducible to Y with:

$$S(Y) = S(Y_1) \cup S(Y_2) \cup \text{----} \cup S(Y_n)$$

It is clear that the class of completely controllable constant rank systems with bounded parameters is the same class of systems considered by Wolovich in Reference (4). Therefore, the following proposition results:

Proposition II.1

For the constant rank system given by (1), if the system is completely controllable and $|\det[\underset{\sim}{W}_n(t)]| > 0$ for all $t \in [t_o, \infty]$, then there exists a state variable feedback matrix F(t), which is defined and bounded for all $t \in [t_o, \infty]$, such that [A(t) + B(t) F(t)] is reducible to a constant matrix Y whose eigenvalues can be arbitrarily specified, where $\underset{\sim}{W}_n(t)$ is the lexicographic representation of $W_n(t)$ (see Reference (4)).

Before giving the necessary and sufficient condition for stabilizing this class of systems the following definitions are presented:

Definition II.3

The system (1) is said to be stabilizable if there exists a matrix valued function F(t) such that if [A(t) + B(t) F(t)] is reducible to Y, then all the eigenvalues of Y will be in the open left half complex plane.

Definition II.4

If the matrix valued function A(t) is reducible to Y_1 and for every matrix valued function F(t) such that [A(t) + B(t) F(t)] is reducible to Y_2 (Y_2

depends on F(t)), there exists a λ such that:

$$\lambda \varepsilon S(Y_1) \cap S(Y_2)$$

then A(t) is said to have an uncontrollable characteristic mode or simply an uncontrollable mode.

Definition II.5

If the matrix valued function A(t) of the system (1) is reducible to Y and Y has an unstable eigenvalue, then A(t) is said to have an unstable mode.

Definition II.6

A(t) is said to have a mode reducible to λ if A(t) is reducible to Y with $\lambda \varepsilon S(Y)$.

This means that an uncontrollable mode of A(t) is an unchangeable characteristic of A(t), and no matter how F(t) is chosen, [A(t) + B(t) F(t)] will still have this same characteristic. This can be interpreted as the reduced versions of A(t) and [A(t) + B(t) F(t)] will have one eigenvalue in common regardless of how F(t) is chosen as long as [A(t) + B(t) F(t)] is reducible.

Proposition II.2

For the constant rank system given by (1), if the following are true:

(i) rank $W_n(t) = p < n$ for all $t \varepsilon [t_o, \infty]$

(ii) $0 < |\det [\tilde{W}_p(t)|$ for all $t \varepsilon [t_o, \infty]$

(iii) the uncontrollable modes are reducible

then there exists a matrix valued function F(t) such that F(t) is defined and bounded for all $t \varepsilon [t_o, \infty]$, and [A(t) + B(t) F(t)] is reducible to a constant matrix Y for which (n-p) eigenvalues are fixed and cannot be arbitrarily specified.

Theorem II.1

Consider the constant rank system (1):

$$\dot{x}(t) = A(t)\, x(t) + B(t)\, u(t)$$

Assume that A(t) and B(t) are both bounded for all t, the uncontrollable modes are reducible and $0 < |\det [\tilde{W}_p(t)]|$ for all $t \varepsilon [t_o, \infty]$, where p is the rank of $W_n(t)$. Let Σ be any set of complex numbers. If A(t) is reducible to Y, then there exists a matrix valued function F(t) such that [A(t) + B(t) F(t)] is reducible to Y with

$$\Sigma \cap S(Y) = \emptyset$$

if and only if there does not exist any uncontrollable mode of A(t) which is reducible to any λ in the set $\Sigma \cap S(Y)$.

Corollary II.2 (Stabilization Theorem)

If the assumptions of Theorem II.1 are satisfied, the system (1) is stabilizable if, and only if, every unstable mode of A(t) is not uncontrollable.

Remark:

In the previous corollary a necessary and sufficient condition for stabilizing a certain class of linear time-varying continuous systems is given. One class of linear systems which falls under this category is the class of linear periodic systems with bounded parameters. Note also that the assumption that the uncontrollable modes are reducible is just for convenience and it is not necessary for the stabilization theory. This is due to the fact that the uncontrollable modes are the unchangeable characteristics of the system. Accordingly, if the part of the system that contains all the uncontrollable modes is stable, the system will be stabilizable regardless of whether or not this part is reducible. The reason for the addition of this assumption is to be able to say that the closed loop system, as a whole, is reducible.

III. SYSTEMS WITH DELAYS

Ross, in Reference (5), gave a sufficient condition for the existence of a feedback control for a system having a time delay in the state only, such that a certain quadratic performance index has a finite value. This automatically implies that this condition is a sufficient condition for stabilizing linear time-invariant systems with time delay in the state only. In Reference (6), another sufficient condition for stabilizing such systems is given. In this section, a necessary and sufficient condition for stabilizing the class of time invariant systems with delays in the state and control, is given. This class contains the class of systems considered in References (5) and (6).

The class of systems to be considered here is described by the following delayed differential matrix equations:

$$\dot{x}(t) = A_1 x(t) + A_2 x(t-h) + B_1 u_1(t) + B_2 u_2(t-h) \qquad (2)$$

$$x(t) = g(t) \qquad t_o - h \leq t \leq t_o$$

$$g(t) \varepsilon C^1 [t_o - h, t_o]$$

where $x(t) \varepsilon R^n$ is the state vector and $u_i(t) \varepsilon R^{r_i}$, $i = 1,2$, are the control vectors. The matrices A_1, A_2, B_1 and B_2 are constant matrices of dimensions compatible with the vectors x, u_1, u_2. The value $h > 0$ represents the delay in the system and it is assumed that all the delays are the same.

To study the regulation of system (2) by static state feedback, let:

$$u_1(t) = F_1 x(t) \qquad (3.a)$$

$$u_2(t) = F_2 x(t) \qquad (3.b)$$

Substituting in (2) gives:

$$\dot{x}(t) = (A_1 + B_1 F_1)\, x(t) + (A_2 + B_2 F_2)\, x(t-h) \qquad (4)$$

and the closed loop characteristic equation will be:

$$\det [A_1 + B_1 F_1 + e^{-sh} (A_2 + B_2 F_2) - sI] = \prod_{i=1}^{n} (s + a_i + b_i e^{-sh}) = 0 \qquad (5)$$

Definition III.1

The system (2) is said to be stabilizable if and only if there exist matrices F_1 and F_2 such that all the roots of (5) will be in the open left half complex plane.

Definition III.2

The system (2) is said to have an uncontrollable characteristic root λ if, and only if, for any choice of F_1 and F_2 of appropriate dimensions,

$$\det [A_1 + B_1 F_1 + e^{-\lambda h} (A_2 + B_2 F_2) - \lambda I] = 0$$

The main theorem of this section is given next.

Theorem III.1 (Stabilization Theorem)

The system:

$$\dot{x}(t) = A_1 x(t) + A_2 x(t-h) + B_1 u_1(t) + B_2 u_2(t-h)$$

$$x(t) = g(t),\ t_o - h \le t \le t_o,\ g(t)\ \varepsilon C^1\ [t_o - h,\ t_o]$$

is stabilizable if, and only if, the following are true:

(i) the uncontrollable characteristic roots are stable

(ii) the uncontrollable characteristic roots with respect to $u_1(t)$ alone with $A_2 = 0$ have real parts less than $1/h$.

Remark:

Condition (ii) means the following: For every λ such that:

$$\lambda \varepsilon \ \{\lambda;\ \lambda \varepsilon S(A_1 + B_1 F_1) \text{ every } F_1\},$$

$\mathrm{Re}(\lambda) < 1/h$.

Remark:

To apply Theorem III.1 the set of uncontrollable characteristic roots must be found. Hence one must multiply the right hand side of (5) for general a_i's and b_i's (since one cannot determine a priori which ones are fixed and which ones are arbitrary), find the determinant on the left hand side for general F_1 and F_2, and then equate the similar coefficients on both sides. This gives a set of simultaneous nonlinear equations which must be solved for the a_i's and b_i's.

It is clear that this procedure is difficult especially in the case of high order systems, except for every special cases. Therefore, another procedure must be found to check the stabilizability properties of systems of the form of (2). Fortunately, as discussed in Reference (9), equations of the form of (5) have only a finite number of unstable roots. Therefore, it is possible to check the stabilizability of the system before going through the above procedure. This can be done as follows: the unstable characteristic roots are found first by using the method given in Reference (9), (these roots represent the unstable roots of the autonomous system that have to be stabilized). Substitute each of these roots into Equation (5) for general F_1 and F_2. If the left hand side vanishes for any of these roots, the system is not stabilizable. With regard to condition (ii) of the theorem, the set of uncontrollable roots with respect to $u_1(t)$ with $A_2 = 0$ can be readily found (see Reference (1)), and hence the condition can be checked. Therefore, it is possible to recognize whether or not the system is stabilizable before going through the above procedure. It should also be noted, however, that finding the unstable roots of the autonomous system is not a trivial one and it requires considerable computational efforts.

IV. SPECIAL CASES

The results given in Sections II and III can be specialized to give a necessary and sufficient condition for stabilizing linear time-invariant systems. For example, if h=0 in Section III, the second condition of Theorem III.1 will be satisfied trivally and hence can be omitted. Therefore the following two corollaries can be stated directly:

Corollary IV.1

Consider the linear time-invariant system:

$$\dot{x}(t) = A\,x(t) + B\,u(t) \tag{6}$$

Let Σ be any non-empty set of complex numbers; then there exists a matrix F such that:

$$S(A + BF) \cap \Sigma = \emptyset$$

if and only if $\lambda \epsilon S(A) \cap \Sigma$ is not controllable.

Corollary IV.2

The system given in Corollary IV.1 is stabilizable if, and only if, every unstable eigenvalue of the matrix A is not uncontrollable. For more details see Reference (1).

Remark:

One of the most important special cases of the results of Section III which has special features is the case when $A_2 = 0$, namely, the delay is only in the control function. In this case Equation (2) reduces to:

$$\dot{x}(t) = Ax(t) + B_1\, u_1(t) + B_2\, u_2(t-h) \tag{7}$$

$$x(t) = g(t) \qquad t_o - h \leq t \leq t_o$$

and Equation (4) will be:

$$\dot{x}(t) = (A + B_1\, F_1)\, x(t) + B_2\, F_2\, x(t-h) \tag{8}$$

and the closed loop characteristic equation will be:

$$\det [A + B_1 F_1 + e^{-sh} B_2 F_2 - sI] = 0 \qquad (9)$$

Note here that the open loop system (7) is a finite dimensional system. Now if the feedback controls given by (3) are applied, then the closed loop system (8) will, in fact, be infinite dimensional. This is because the control $u_2(t)$ is a delayed control, and accordingly, if it is a feedback control, it is a function of the delayed state. Thus the resultant closed loop system will have a delay in the state as given by Equation (8). The feedback control $u_2(t)$, $t_o-h \leq t \leq t_o$, is a function of the initial state $x(t) = g(t)$, $t\varepsilon [t_o-h, t_o]$, which is assumed to be known.

To illustrate an important characteristic of this class of system, assume that $B_1 = 0$. Now if λ is an uncontrollable characteristic root, then

$$\det [A + e^{-\lambda h} B_2 F_2 - \lambda I] = 0$$

For all matrices F_2 of appropriate dimensions. This is true also for $F_2 = 0$. Thus:

$$\det [A - \lambda I] = 0$$

This means that λ is an eigenvalue of A. Since A is a nxn matrix, the system (7) has at most n uncontrollable characteristic roots. This is due to the fact that the delay is only in the control and therefore this system is a finite dimensional system. Accordingly, if there is any fixed property of this system that is unchangeable by state feedback, it must have the same nature of finite dimensionality since it belongs to the open loop system. So, even though Equation (9) has infinitely many solutions, there exists only a finite number of uncontrollable characteristic roots.

The set of uncontrollable characteristic roots can be easily characterized in this case. For more details, see Reference (1).

V. CONCLUSION

In this study, the problem of stabilizing two types of linear continuous systems has been investigated. The problem has been studied for the class of constant rank systems with bounded parameters, and the class of time-invariant systems with time delays in the state and control. A necessary and sufficient condition for stabilizing constant rank systems with bounded parameters was given. This result extends the work of Wolovich, Reference (4), which was done for controllable systems, to cover systems that are not necessarily controllable. It was shown that the reducibility property of a class of time varying systems plays an important role in the analysis of such problems.

For the problem of stabilizing time-invariant systems with time-delays, a necessary and sufficient condition was given. This class of systems contains, as a special case, the class of systems considered in References (5) and (6) for which sufficient conditions for stabilization were given.

Also it was shown that the results given here can be specialized to give a necessary and sufficient condition for stabilizing ordinary time-invariant systems, which is equivalent to that given by Wonham in Reference (2).

VI. ACKNOWLEDGEMENT

The research reported herein was performed in the Control Systems Science and Engineering Program at Washington University, St. Louis, Missouri, and was sponsored in part by the Air Force Office of Scientific Research under AFOSR Contract No. F-44620-69-C-0116 and AFOSR Grant No. 73-2496A, and by the National Science Foundation under Grant No. 57329A.

REFERENCES

1. El-Geneidy, M. M., "Stabilization of Linear Continuous Systems: An Algebraic Approach", D.Sc. Dissertation, Washington University, St. Louis, Missouri, May 1974.
2. Wonham, W. M., "On Pole Assignment in Multi-input Controllable Systems", IEEE Transactions on Automatic Control, Vol. AC-12, No. 6, December 1967, pp. 660-665.
3. Silverman, L. M. and Meadows, H. E., "Equivalent Realization of Linear Systems", SIAM Journal on Applied Mathematics, Vol. 17, No. 2, March 1969, pp. 393-408.
4. Wolovich, W. A., "On the Stabilization of Controllable Systems", IEEE Transactions on Automatic Control, (short paper), Vol. AC-13, No. 5, October 1968, pp. 569-572.
5. Ross, D. W., "Controller Design for Time Lag Systems via a Quadratic Criterion", IEEE Transactions on Automatic Control, Vol. AC-16, No. 6, December 1971, pp. 664-672.
6. Nazaroff, G. H., and Hewer, G. A., "Stabilization of Linear Autonomous Differential Delay Systems", IEEE Transactions on Automatic Control (Technical Notes and Correspondence) Vol. AC-18, No. 6, December 1973, pp. 673-674.
7. Brockett, R. W., Finite Dimensional Linear Systems, John Wiley and Son, Inc. New York, 1970.
8. Silvermann, L. M., "Realization of Linear Dynamical Systems", IEEE Transactions on Automatic Control, Vol. AC-16, No. 6, December 1971, pp. 554-567.
9. Bellman, R. and Cooke, K. L., Differential-Difference Equations, Academic Press, New York, 1963.

ON THE STABILIZATION OF A CLASS OF NONLINEAR SYSTEMS VIA HYPERSTABILITY

TUDOR IONESCU* AND R.V. MONOPOLI*
Department of Electrical and Computer Engineering
University of Massachusetts, Amherst, Massachusetts

ABSTRACT

Sufficient conditions for asymptotic feedback stabilization of both continuous and discrete bilinear systems are given using a hyperstability type technique.

I. INTRODUCTION

In this note we consider the problem of feedback stabilization for continuous and discrete bilinear systems. Related but less general problems have been studied in [1].

Displaying better step and sinusoidal response performances than linear controllers ([2], pp. 201), the bilinear controllers appear to have important applications in adaptive control problems. A specific development in this direction is the object of a future paper. Here we merely consider the following situation. Let:

$$\dot{x} = \left(\sum_{i=1}^{m} u_i(t)\, A_i\right) x + bv \qquad (1)$$

$$y = C^T x \qquad (2)$$

where: $A_i \in R^{nxn}$ b, $C \in R^n$, v(t), $u_i(t)$ being real scalar continuous function with A_i, b, C time invariant.

Problem: Given A_i, b determine C and the feedback controls $u_i = u_i(x)$, $v=v(x)$ such that (1),(2) is asymptotically stable for all $||x_o|| \leq M < \infty$, for some M.

Remark: The system (1),(2) is intrinsic bilinear i.e. not all $u_i \equiv 0$ for $x \neq 0$. The discrete version of the above problem is also considered in section 2.

1. Continuous Case

For the system (1),(2) we have the following Proposition:

Proposition 1:

If for some negative definite matrix P and for some $i=i_1 \varepsilon \{1,\ldots,m\}$,

$$u_{i_1}(x) = x^T(A_{i_1} P + P A_{i_1}{}^T)x + C^T x \text{ , } u_i = 0,\ i \neq i_1 \qquad (3)$$

$$v(x) = x^T(A_{i_1} P + P A_{i_1}{}^T)x - C^T x \qquad (4)$$

$$C^T = -\frac{2}{3} b^T P \quad \text{or} \qquad (5a)$$

$$C^T = -2b^T P \qquad (5b)$$

---

*This work was supported by NASA NGR-22-010-018.

then there exist the constants β_1,β_2 such that $\beta_1\ ||x(t_1)||^2 \leq \beta_2\ ||x(t_o)||^2$ for $\forall t_1 \geq t_o$, $\beta_1, \beta_2 > 0$. (*)

Moreover if (A_{i_1}, b) is a controlable pair and

$$\{x\varepsilon R^n | x^T(A_{i_1} P + P A_{i_1}^T)x=0\} \cap \{x\varepsilon R^n | b^T Px=0\} = \{0\} \tag{6}$$

and

$$\{x\varepsilon R^n | x^T(A_{i_1} P + P A_{i_1}^T)x - C^Tx=0\} \cap \{x\varepsilon R^n | A_{i_1} x=0\} = \{0\} \tag{6'}$$

then

$$\lim_{t_1\to\infty} ||x(t_1)||=0 \tag{**}$$

Proof:

Without loss of generality one can take $t_o=0$. We associate with (1), (2) the integral:

$$\eta(0,t_1) = \int_o^{t_1} (v - \sum_{i=1}^m u_i)\ C^Tx \cdot dt,\ t_1 \geq 0 \tag{7}$$

Clearly for some symmetric matrix P:

$$\eta(0,t_1) = \int_o^{t_1} \{(v - \sum_{i=1}^m u_i)\ C^Tx + \frac{d}{dt} x^TPx - \frac{d}{dt} x^TPx\}\ dt =$$

$$= \int_o^{t_i} \{x^T \sum_{i=1}^m u_i (A_iP + A_i{}^TP)x + 2vb^TPx + vC^Tx - \sum_{i=1}^m u_i \cdot C^Tx\}dt$$

$$- x^T(t_1)Px(t_1) + x^T(0)\ Px(0) \tag{8}$$

Denote by $Q_i = A_iP + P\ A_i{}^T$. (8) becomes:

$$\eta(0,t_1) = \int_o^{t_1} \{\sum_{i=1}^m u_i(x^TQ_ix - C^Tx) + v(2b^TP + C^T)x\}\ dt$$

$$- x(t_1)Px(t_1) + x^T(0)\ Px(0) \tag{9}$$

Replacing (3)-(5a) in (9) one obtains:

$$\eta(0,t_1) = \int_o^{t_1} \{[(x^TQ_{i_1}\ x) - C^Tx]^2\}\ dt -$$

$$- x^T(t_1)Px(t_1) + x^T(0)Px(0). \tag{10}$$

From (3),(4),(7),(10) and letting $P^T = P < 0$, one obtains:

$$\beta_1||x(t_1)||^2 \leq \int_o^{t_1}\{-[x^TQ_{i_1}\ x - C^Tx]^2 - 2(C^Tx)^2\}dt + \beta_2||x(0)||^2 \tag{11a}$$

Similarly replacing (3)-(5b) in (9) and using (3),(4),(7) one gets:

$$\beta_1||x(t_1)||^2 \leq \int_o^{t_1} \{-(x^TQ_{i_1}\ x)^2 - (C^Tx)^2\}dt + \beta_2||x(0)||^2 \tag{11b}$$

Clearly (11a) and (11b) imply (*). To show, for instance, that (11b) implies (**) assume the contrary: there exists $\varepsilon > 0$, arbitrarily small such that:

$$||x(t)|| \geq \varepsilon > 0 \text{ for } \forall t \geq t'_1 \tag{12}$$

Since $(x^TQ_{i_1}x)^2 + (C^Tx)^2$ is continuous in x, from (6) and (12) it follows that there exists $\varepsilon' > 0$ such that:

$$(x^TQ_{i_1}x)^2 + (C^Tx)^2 \geq \varepsilon' > 0 \quad \forall t \geq t'_1 \tag{13}$$

(13) clearly implies:

$$(x^TQ_ix)^2 + (C^Tx)^2 \geq \varepsilon' \geq \varepsilon'' \cdot e^{-\alpha t} \cdot M \geq \varepsilon'' \cdot e^{-\alpha t} \cdot ||x(0)||^2 \tag{14}$$

for some $\varepsilon'' > 0$ small, and $\alpha = \frac{\varepsilon''}{\beta_2} > 0$, $\forall t \geq t'_1$.

(14) and (11b) yield:

$$\beta_1||x(t_1)||^2 \leq \int_0^{t_1} \{-\varepsilon'' \cdot e^{-\alpha t} \cdot ||x(0)||^2\}dt + \beta_2||x(0)||^2 \tag{15}$$

(15) gives:

$$\beta_1||x(t_1)||^2 \leq \frac{\varepsilon''}{\alpha} e^{-\alpha t_1} \cdot ||x(0)||^2 - \frac{\varepsilon''}{\alpha}||x(0)||^2 + \beta_2||x(0)||^2 \tag{16}$$

(16) implies $\lim\limits_{t_1 \to \infty} ||x(t_1)|| = 0$ which contradicts (12)

and hence (**) is valid.

Q.E.D.

Remarks:

1. Relation (6) is needed to insure that there are no critical points besides $x \equiv 0$. To see this write (1):

$$x = (u_{i_1}(x)\,A_{i_1})x + bv(x) = x^TQ_{i_1}x(A_{i_1}x+b) + C^Tx(A_{i_1}x-b) = F(x) \tag{17}$$

Clearly (6) implies $F(x)=0$ only if $x \equiv 0$ for if $F(\tilde{x})=0$, $\tilde{x} \not\equiv 0$ then either $\{C^T\tilde{x}=0,\ \tilde{x}^TQ_{i_1}\tilde{x} \neq 0,\ A_{i_1}\tilde{x}=-b\}$ or $\{C^T\tilde{x} \neq 0,\ A_{i_1}\tilde{x}=b,\ \tilde{x}^TQ_{i_1}\tilde{x}=0\}$ or $\{v(\tilde{x})=0,\ A_{i_1}\tilde{x}=0\}$ or $\{u_{i_1}(\tilde{x}) \neq 0,\ v(\tilde{x}) \neq 0,\ (u_{i_1}(\tilde{x})A_{i_1})\tilde{x}=-bv(\tilde{x})\}$.

In the first case $C^T\tilde{x}=0$ implies (using (5a) or (5b) and $A_i\tilde{x} \equiv -b$):

$$\tilde{x}(t^T_1)\,A_{i_1}P\tilde{x}(t_1) = 0 \tag{18}$$

But (18) implies that $\tilde{x}^T(t_1)P\,A_{i_1}^T\,\tilde{x}(t_1) = 0$ and so

$\tilde{x}^T(t_1)(A_{i_1}P + P\,A_{i_1}^T)\tilde{x}(t_1) = 0$ which contradicts (6).

The second case leads to contradiction in a similar way.
The third case contradicts (6)'. Also if $u_{i_1}(\tilde{x})=0$ then $v(\tilde{x}) \neq 0$ and since $b \not\equiv 0$ one has again a contradiction.

Finally if $u_{i_1}(\tilde{x})=0$, $v(\tilde{x})=0$ but

$$(u_{i_1}(\tilde{x})\, A_{i_1})\tilde{x} = -bv(\tilde{x}) \tag{19}$$

then $F(\tilde{x})=0$. But (19) is equivalent with:

$$A_{i_1}\tilde{x} = -b\,\frac{v(\tilde{x})}{u_{i_1}(\tilde{x})} \tag{20}$$

Taking the derivative in (20), dividing by $u(\tilde{x})$ and doing this n times one has:

$$A_{i_1}^2\tilde{x} = -bw(\tilde{x}) - A_{i_1}\, b\frac{v(\tilde{x})}{u_{i_1}(\tilde{x})} \tag{21}$$

$$\vdots \qquad\qquad \vdots$$

$$A_{i_1}^n\tilde{x} = -bw_1(\tilde{x}) - A_{i_1}\, bw_2(\tilde{x}) - \dots - A_{i_1}^{n-1}\, b\,\frac{v(\tilde{x})}{u_{i_1}(\tilde{x})} \tag{22}$$

where w, w_1, w_2 are functions of u, v, and their derivatives. From Cayley-Hamilton formula the LHS in (22) can be expressed as a linear combination of the vectors $A_{i_1}^{n-1}\tilde{x}, \dots, A_{i_1}\tilde{x}$. But since $v(\tilde{x})\neq 0$, $u_{i_1}(\tilde{x})=0$ and (A_{i_1},b) is a controlable pair of RHS of (22) is a vector linear independent of the RHS vectors in (20)-(22). Contradiction.
It follows that Proposition 1 gives global asymptotic stability conditions for the class of systems (1),(2).

2. Let:

$$N_{\tilde{x}} = \{\tilde{x}\in R^n \,|\, u_{i_1}(\tilde{x})\neq 0,\ v(\tilde{x})\neq 0\}$$

Then the controlability of (A_{i_1},b) in Proposition 1 can be replaced by the following conditions:

$$\{N_{\tilde{x}}\} \cap \{\tilde{x}\in R^n \,|\, A_{i_1}\tilde{x} + \alpha b = 0\} = \{0\}$$

for any real finite scalar α i.e. b does not belong to the range of $A_{i_1}\tilde{x}$ for $\forall\tilde{x}\in N_{\tilde{x}}$. An example of such system in which A_{i_1},b is not a controlable pair is given in the last section.

3. It is clear that (10) implies:

$$\beta_1||x(t_1)||^2 \le \eta(0,t_1) + \beta_2||x(0)||^2$$

which asserts the hyperstability property of (1),(2),(7) for the feedback (3),(4),(5), as stated in [3]. The index $\eta(0,t_1)$ defined by (7) corresponds to the relation (50) [3], pp. 351, for β_2, $\beta_3 = 0$ in the special case of bilinear systems.

2. The Discrete Case

Let:

$$x_{k+1} = \left(\sum_{i=1}^{m} u_i(k)\, A_i\right) x_k + bv_k \tag{1$'$}$$

$$y_k = C^T x_k \tag{2$'$}$$

where $A_i \in R^{nxn}$, b, $C\in R^n$, be the discrete version of (1),(2).

Proposition 1': Let P be some negative definite matrix. If $(\sum_{i=1}^{m} A_i)(I-2bb^T P)$ is nonsingular, then: (1) If $\sum_{i,j} A^T_i P A_j \geq 0$ the stabilizing controls are: $\{u_1=u_2=..=u_m=\rho$, with $v_k=0$ or $v_k = -\rho \dfrac{2b^T P(\sum^m A_i)x_k}{b^T Pb}$ for some $\rho\neq 0\}$. (ii) If $\sum_{i,j} A^T_i PA_j < 0$, the stabilizing controls are: $\{v_k=0$ or $v_k = \dfrac{-2b^T P(\sum^m A_i)x_k\cdot u}{b^T Pb}$, $u_1=u_2=..=u_m=u=\dfrac{(x_k R x_k)^{1/2}}{[x_k^T(-\sum_{i,j} A_i^T PA_j)x_k]^{1/2}}$ where $R^T=R>0$ and $R + P<0\}$. (iii) If $\sum_{i,j} A_i^T PA_j \leq 0$ but for some constant $\rho>0$, $\rho\cdot \sum_{i,j}^{m} A_i^T PA_j - P > 0$ the stabilizing controls are: $\{u_1=u_2=..=u_m=\rho^{1/2}$ with $v_k=0$ or $v_k = \dfrac{-2\rho^{1/2}\cdot b^T(\sum^m A_i)x_k}{b^T Pb}\}$

Proof: We associate as before with (1)',(2)' the sum:

$$\eta(0,k_1) = \sum_{k=0}^{k_1} (v_k - \sum_{i=1}^{m} u_i(k))C^T x_k \tag{3'}$$

For some negative definite matrix P one has:

$$\eta(0,k_1) = \sum_{0}^{k_1} \{(v_k - \sum_{i=1}^{m} u_i(k))C^T x_k + (x^T_{k+1} Px_{k+1} - x^T_k Px_k) - x^T_{k+1} Px_{k+1} + x^T_k Px_k\} \tag{4'}$$

Taking $u_1=u_2..=u_m=u$ and $v_k=0$ in (4)' one has:

$$\eta(0,k_1) = \sum_{0}^{k_1} -m_1 u\cdot C^T x_k + u^2\cdot x^T_k(\sum_{i,j} A_i^T PA_j)x_k - x^T_k Px_k\} - x^T_{k_1+1} Px_{k_1+1} + x^T_o Px_o \tag{5'}$$

Now if the condition (i) is satisfied, replacing $u=\rho$ in (5)' and using (3)' one gets:

$$\beta_1 ||x_{k_1+1}||^2 \leq \sum_{k=0}^{k_1} x^T_k Px_k + \beta_2 ||x_o||^2 \leq \beta_2 ||x_o||^2 \tag{6'}$$

Moreover we claim that (6)' implies: $\lim_{k_1\to\infty} ||x_{k_1}|| = 0$ (7)'

Indeed let's assume that it is false. Then there exists $\varepsilon > 0$ arbitrarily small and k_1' such that $||x_k|| \geq \varepsilon > 0$ $\forall k \geq k_1'$ (8)'

It follows:

$$x^T_k(-P)x_k \geq \varepsilon' \geq \varepsilon'' \alpha^k \cdot M \geq \varepsilon'' \cdot \alpha^k \cdot ||x_o||^2 \tag{9'}$$

where $\varepsilon'' > 0$ small, $\alpha = 1 - \frac{\varepsilon''}{\beta_2} < 1$ and $||x_o||^2 \leq M < \infty$ for some M.

$$\beta_1 ||x_{k_1+1}||^2 \leq \sum_{k=0}^{k_1} - \varepsilon'' \cdot \alpha^k ||x_o||^2 + \beta_2 ||x_o||^2 \tag{10'}$$

If $k_1 \to \infty$ (10)' gives: $\lim_{k_1\to\infty} ||x_{k_1+1}||^2 \leq -\varepsilon'' \frac{1}{1-\alpha} ||x_o||^2 + \beta_2 ||x_o||^2 = 0$

which contradicts (8)' concluding the proof of the case (i). The cases (ii), (iii) follow in similar fashion.

Q.E.D.

Remark: The assumption on the matrix $(\sum_{i}^{m} A_i)(I-bb^T P)$ is needed to insure that there are no critical points besides $x \equiv 0$, as can be easily checked.

Example: Let:

$$\dot{x} = u \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix} x + \begin{vmatrix} 0 \\ 1 \end{vmatrix} v \ , \quad y = C^T x$$

Choosing $P = -\frac{1}{3} I$ (the identity matrix) one has (Proposition 1):

$$u(x) = -\frac{2}{3}(x_1^2 + x_2^2) + \frac{2}{3} x_2 \ , \quad v(x) = -\frac{2}{3}(x_1^2 + x_2^2) - \frac{2}{3} x_2$$

To verify the claim of asymptotic stability consider a Lyapunov function $V = x_i^2 + x_2^2$. One can immediately check that:

$$\dot{V} = -\frac{4}{3} x_1^2 (x_1^2 + x_2^2) - \frac{4}{3} x_2^2 (x_1^2 + x_2^2) - \frac{4}{3} x_2^2 < 0$$

BIBLIOGRAPHY

[1] Wood, Johnathan, Ph.D. dissertation, "Power Conversion in Electrical Networks," Chapter 4, April 1973, M.I.T., Cambridge, Mass.

[2] Mohler, R.R., Rink, R., "Control with multiplicative mode," ASME Transactions, Series D, 91, 1969, pp. 201.

[3] Popov, V.M., "Incomplete controlable positive systems," Revue Roumaine des Sciences Technique, serie Electrotechnique et Energetique, TOME 12, 1967, pp. 337.

REGION OF STABILITY FOR NONLINEAR DIFFERENCE EQUATIONS

A. S. C. SINHA
Department of Electrical Engineering and Mathematics
Indiana Institute of Technology
Fort Wayne, Indiana 46803

ABSTRACT

The problem of stability properties for the solutions of nonlinear difference equations is considered. In this paper we shall show some results which give explicit regions of stability. The objective of this paper is to replace these words "sufficiently small" by an explicit estimate which is simple.

INTRODUCTION

Equations of the form

$$x(n+1) = A(n)\ x(n) + f(n,x(n)) \tag{1}$$

have become important in sampled data systems and related circuit problems. As for the applications of Lyapunov functions to the stability problems of difference equations with discrete variables, one can find some results in [1, 7]. Recent works on the problem of the absolute stability of sampled-data systems for obtaining stability for certain class of systems can be found in [8, 10]. The main objective in this paper is to provide an explicit region of stability of system (1). In earlier papers, no explicit region for stability has been shown.

LEMMA 1.

Let $\phi\ (n,n_0)$ be a fundamental matrix of

$$x(n+1) = A(n)\ x(n) \tag{2}$$

such that $\phi(n_0,n_0) = I$, then there exists a constant matrix $\tilde{P}$ if and only if the mean value

$$\text{(i)} \quad \tilde{P} = \lim_{N\to\infty} \frac{1}{2N} \sum_{n=-N}^{N} \phi^T(n,s)\ Q\phi(n,s)\ \Delta n$$

exists boundedly, is nonsingular for some positive definite matrix Q and independent of n such that $\tilde{P} = P$. The linear system (2) is then asymptotically stable if

$$\text{(ii)} \quad A^T(n)\, P\, A(n) - P < 0$$

Proof.

If P exists and is constant, then $\phi^T(n,s)\, P\, \phi(n,s) = P$ and $\tilde{P} = P$.

If $\tilde{P}$ exists and is independent of n, then

$$\phi^T(s,n_0)\ \tilde{P}(s)\ \phi(s,n_0) = \lim_{N\to\infty} \frac{1}{2N} \sum_{n=-N}^{N} \phi^T(n,n_0)\ Q\ \phi(n,n_0)\ \Delta n$$

$$= \tilde{P}(n_0)$$

Since $\phi^T(s,n_0)\ \tilde{P}(s)\ \phi(s,n_0) = \tilde{P}(n_0)$ for all n_0 and s, $\tilde{P}$ is clearly symmetric and nonnegative constant matrix but it is assumed nonsingular so it must be positive. It can easily be shown that the linear system is stable if the condition (ii) is satisfied.

Lemma 1. is a generalization of a result given by Sinha [11].

Theorem 1.

Suppose

$$|f(n,x(n)| < \mu\sqrt{\frac{\Delta}{\lambda}}\ \|x(n)\|,\ \mu = \min\left(1,\ \frac{\delta}{\lambda(2\alpha+1)}\right),\ 0 < \alpha < 1$$

and Lemma 1. holds. Furthermore, if there exists a constant $\rho > 0$ such that $\|x(n,n_0,x_0)\| < \rho$ then for any $\varepsilon < \rho\sqrt{\frac{\Delta}{\lambda}}$, and any initial function $\|x(n_0,n_0,x_0)\| < \varepsilon$ implies $\|x(n,n_0,x_0)\| < \varepsilon\sqrt{\frac{\Delta}{\lambda}}$ for all $n > n_0$.

Proof.

For any vector $x(n;n_0,x_0)$, there exists a constant matrix $\tilde{P} = P$ such that

$$\Delta|x(n;n_0,x_0)|^2 \leq x^T(n;n_0,x_0)\ P\ x(n;n_0,x_0) \leq \lambda|x(n;n_0,x_0)|^2 \tag{3}$$

Now suppose that, for some solution $x(n;n_0,x_0)$; $\|x(n;n_0,x_0)\| < \varepsilon < \rho\sqrt{\frac{\Delta}{\lambda}}$. Then $|x(n;n_0,x_0)| < \varepsilon$ for some $n_0 \leq n \leq N_0$ so that using (2) $x^T(n;n_0,x_0)\ P\ x(n;n_0,x_0) < \lambda\varepsilon^2$. That is, the solution trajectories lie within the quadratic surface $S(n) = x^T(n;n_0,x_0)\ P\ x(n;n_0,x_0) = \lambda\varepsilon^2$ during the interval $[n_0,N_0]$. We wish to show that such a solution remains in this surface $S(n)$ for all $n \geq N_0$.

To prove this assertion, let the trajectory $X(n) = x(n;n_0,x_0)$ reach the surface $S(n)$ for the first time when $n = n_1$; $n_1 \geq n_0$. Then

$$x^T(n_1;n_0,x_0)\ P\ x(n_1;n_0,x_0) = \lambda\varepsilon^2$$

and using (3) we have

$$|x(n_1;n_0,x_0)| \geq \varepsilon.$$

However, by assumption

$$x^T(n_1;n_0,x_0)\ P\ x(n;n_0,x_0) < \lambda\varepsilon^2 \quad \text{for} \quad n_0 \leq n < n_1$$

Hence, by using (3),

$$|x(n;n_0,x_0)| < \varepsilon\sqrt{\frac{\lambda}{\Delta}} \quad \text{in} \quad n_0 \leq n < n_1$$

Also,

$$\begin{aligned} \Delta V &\triangleq V(n+1) - V(n) \\ &= X^T(n)\left[A^T(n)\ P\ A(n) - P\right] X(n) \\ &\quad + 2P\ A(n)\ X(n)\ f + P\ f^2 \end{aligned} \tag{4}$$

With the assumptions of Lemma 1., the system $x(n+1) = A(n)\,x(n)$ is asymptotically stable implying

$$\|A(n)\| < \alpha \ ; \quad 0 < \alpha < 1$$

Since

$$|P\,X(n)| \leq \lambda |X(n)| \ ; \quad |f| < \min\left(1, \frac{\delta}{\lambda(2\alpha+1)}\right) \sqrt{\frac{\Delta}{\lambda}}\ \|X(n)\|, \ \sqrt{\frac{\Delta}{\lambda}} \leq 1.$$

Therefore,

$$\begin{aligned} \Delta V &\leq -\,\delta |X(n)|^2 + 2\alpha\lambda |X(n)|\ |f| + \lambda |f|^2 \ ; \quad \delta > 0 \\ &\leq -\ |X(n)|\ [\delta |X(n)| - \lambda |f|\ (2\alpha+1)] \end{aligned}$$

Since $\frac{\delta}{\lambda(2\alpha+1)} \leq 1$ by choice of λ. Therefore, on setting $|X(n_1)| \geq \varepsilon$ gives $\Delta V < 0$ at $n = n_1$, so that the trajectory cannot leave $S(n)$ and is confined by the surface for all $n \geq n_0$, which proves the Theorem.

The following lemma may be proved by means of the same method as in Halanay [6].

Lemma 2.

Let $\phi(n,n_0)$ be a fundamental matrix of the homogeneous system $x(n+1) = A(n)\,x(n)$ such that $\phi(n_0,n_0) = I$ (the unit matrix). If the trivial solution of it is uniformly asymptotically stable, then there exists positive constants B and α such that

$$\|X(n)\| \leq B \exp\,(-\alpha(n - n_0)) \quad \text{for any } n \geq n_0.$$

Theorem 2.

Suppose

$$|f| < \mu\, \|x(n;n_0,x_0)\|$$

and if there exists a constant $\rho > 0$ such that $\|x(n;n_0,x_0)\| < \rho$, then for any $\varepsilon < \rho\sqrt{\frac{\Delta}{\lambda}}$, and any initial function $\|x(n_0,n_0,x_0)\| < \varepsilon$ implies $\|x(n,n_0,x_0)\| \to 0$ as $n \to \infty$.

Proof.

Suppose for a solution $x(n;n_0,x_0)$, $\|x(n_0;n_0x_0)\| < \varepsilon\sqrt{\frac{\lambda}{\Delta}} \leq \rho$ for all $n \geq n_0$. We claim such a solution satisfies an estimate

$$\|x(n;n_0,x_0)\| \leq \left(\frac{2B\mu}{1-c}\right)^K \rho \quad \text{when } n \geq n_K = n_0 + K\,N \tag{5}$$

where $c = e^{-\alpha} < 1$, $N = n_1 + \frac{1}{\alpha} \log \left(\frac{\mu}{1-c}\right)$, $K = 0, 1, \ldots$. The proof follows from induction.

When $K = 0$, (5) gives $\|x(n;n_0,x_0)\| \leq \rho$ which is true. Now suppose then (5) is true for $K = M$.

$$\|x(n;n_o,x_o)\| \leq \left(\frac{2B\mu}{1-c}\right)^M \rho \quad \text{for } n \geq n_M \tag{6}$$

Now we will show that the inequality (5) is also true for K = M + 1. One easily obtains, following lemma 2., that the solution $x(n;n_o,x_o)$ satisfies the estimate

$$\begin{aligned}
|x(n,n_o,x_o)| &\leq B\, e^{-\alpha(n-n_o)}\, |x(n_o)| \\
&\quad + B \sum_{s=n_o}^{n-1} \left(\frac{1}{e^{\alpha}}\right)^{(n-s-1)} \|f(s,x(s,n_o,x_o))\| \\
&\leq B\, e^{-\alpha(n-n_o)}\, |x(n_o)| \\
&\quad + B \sup_{f(s)\; s\geq 0} \|f(s,x(s,n_o,x_o))\| \frac{1-\left(\frac{1}{e^{\alpha}}\right)^n}{1-\left(\frac{1}{e^{\alpha}}\right)} \\
&< B\, e^{-\alpha(n-n_o)}\, |x(n_o)| + B \left(\frac{2B\mu}{1-c}\right)^M \rho\, \mu \left(\frac{1}{1-c}\right)
\end{aligned}$$

where $c = \frac{1}{e^{\alpha}} < 1$. $|f| \leq \mu \|x(n;n_o,x_o)\|$. Therefore,

$$\begin{aligned}
|x(n;n_o,x_o)| &\leq B\, e^{-\alpha(n-n_o)}\, |x(n_o)| + \left(\frac{2B\mu}{1-c}\right)^{M+1} \frac{\rho}{2} \\
&\leq \left(\frac{2B\mu}{1-c}\right)^{M+1} \rho \quad \text{for } n \geq n_M + \frac{1}{\alpha} \log \left(\frac{\mu}{1-c}\right)
\end{aligned}$$

The last step follows by reducing the first term equal to the magnitude of the second term when $n \geq n_M + \frac{1}{\alpha} \log \left(\frac{\mu}{1-c}\right) = n_{M+1}$

Since $\left(\frac{2B\mu}{1-c}\right) < 1$ by choice of μ.

Therefore

$$\lim_{n\to\infty} \|x(n;n_o,x_o)\| = 0$$

This completes the proof of the Theorem 2.

REFERENCES

1. Vidal, P., "Non-linear Sampled-Data Systems," Gordon and Breach (1969).
2. Kalman, R. E. and Bertram, J. E., "Control System Analysis and Design Via the 'Second Method' of Lyapunov, II, Discrete-time Systems," Trans. ASME, Ser. D., J. Basic Engrg. 82 (1960), 394-400.
3. Hale, J. K., "Asymptotic Behavior of the Solutions of Differential-Difference Equations," RIAS Technical Reports, 1961, Baltimore 12, Maryland.
4. Gordon, S. P., "Stability and Summability of Solutions of Difference Equations," Mathematical Systems Theory, Vol. 5, No. 1, (1971) pp. 56-65.
5. Hahn, W., "Theory and Application of Liapunov's Direct Method," Prentice Hall, Englewood Cliffs, N. J., (1963).
6. Halanay, A., "Quelques Questions de la Theorie de la Stabilite pour les Systemes aux Differences Finies," Arch. Rational Mech. Anal. 12 (1963), 150-154.
7. Puri, N. N. and Drake, R. L., "Stability Studies of a Class of Non-linear Difference Equations Using Liapunov's Second Method," J. Franklin Inst., Vol. 279, No. 3, (1965) pp. 209-217.
8. Tsypkin, Y. Z., "Absolute Stability of Sampled-Data Control Systems," Dokl. Akad. Nauksssr 145, pp. 52-55.
9. Szego, G. P., "On the Absolute Stability of Sampled-Data Control Systems," Proc. Nat. Acad. of Sciences, Sept. 1963.
10. Szego, G. P. and Kalman, R. E., "Sur la Stabilite Absolue d'un Systeme d'equations aux Differences Finies," Comptes, Rend. Acad. Sci. (Paris), 257, 388-390 (1965).
11. Sinha, A. S. C., "On the Lagrange Stability and Boundedness of Difference Equations," Int. J. Control, (To appear).

SUSTAINED OSCILLATIONS FOR A CERTAIN CLASS OF $n^{\underline{th}}$ ORDER NONLINEAR SYSTEM

KISHAN L. DUDKIKAR
The MITRE Corporation
McLean, Virginia

ABSTRACT

This paper presents an approach to finding the limit cycle amplitude and frequency of oscillation for a certain class of $n^{\underline{th}}$ order nonlinear system. This approach, which uses the phase margin and gain margin criteria, has not been explicitly investigated. This paper shows that the solution for this type of nonlinear system reduces to the determination of the roots of a polynomial of order n. This paper also shows that, by removing some of the poles of the open loop transfer function, the limit cycle conditions are determined from examination of a lower order nonlinear system.

An example of a third order control system in the presence of a multi-valued type nonlinearity is studied. Finally, this paper determines limit cycle stability for this example.

INTRODUCTION

Since a limit cycle, once excited, will sustain oscillation in the absence of further excitation and even in the presence of a sinusoidal input, determination of the limit cycle is absolutely necessary for understanding proper system performance. Therefore an analytical tool to study limit cycles is required.

Several methods of solving differential equations with some specific nonlinearities have been developed. However, as the order of the linear frequency response function involving nonlinear elements in the closed loop transfer function increases, determination of the limit cycle becomes complicated. Investigations have indicated that the phase-plane method is very practical only up to third order control systems[2]. For control systems of higher order, the describing function (DF) method is often used. This method improves in accuracy with higher order systems and thus is complimentary to the phase-plane technique.

It is necessary to point out that the phase margin and gain margin technique plays an important role in determining the time response of high order nonlinear control systems. This technique gives information about the stability of nonlinear systems and about the possible existence of limit cycles in their outputs.

Using the phase margin/gain margin method basically several curves of the loci of the linear transfer fucntion and of the negative inverse of the DF for the nonlinearity of the closed loop control system are drawn on a complex plane. If the curves drawn above intersect, then the system will be capable of exhibiting a limit cycle, stable or unstable. That is, for a limit cycle to occur, at least one of the roots must be on jw axis. The amplitude and frequency of oscillation at the limit cycle is found from the point of intersection.

DEVELOPMENT OF CONDITIONS FOR SUSTAINED OSCILLATIONS

Consider a closed loop (feedback) control system of order n involving nonlinear element as shown in Figure 1. G(s) for this system is represented by:

$$G(s) = \frac{K}{s\ (1 + sT_1)\ (1 + sT_2)\ \dots\ (1 + sT_{n-1})} \qquad (1)$$

where K, T_1, ..., T_{n-1} are constants. The nonlinearity of the system is now replaced by its DF, N(E), where $e = E \sin \omega t$ is the input to the nonlinearity. In general, the DF is a function of both the amplitude, E, and frequency, ω. Here, it is assumed to depend only on E. N(E) is written as:

$$N(E) = a(E) + j\, b(E) \tag{2}$$

From the phase margin and gain margin technique one can identify the conditions for sustained oscillations of the system illustrated in Figure 1 as:

$$|N(E)\, G(j\omega)| = 1 \qquad \text{Gain Margin} \tag{3}$$

$$\angle N(E) + \angle G(j\omega) = -(1 + 2n)\,\Pi \qquad \text{Phase Margin} \tag{4}$$

where n is the order of the control system, and $s = j\omega$.

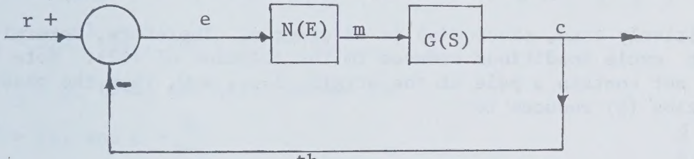


FIGURE 1 - NONLINEAR $n^{\underline{th}}$ ORDER CONTROL SYSTEM

For brevity, the variables a(E) and b(E) are written as a and b respectively unless otherwise stated.

Substituting (1) and (2) into (3) and (4), one obtains:

$$K^2\,(a^2 + b^2) = \omega^2 \prod_{i=1}^{n-1} (1 + \omega^2 T_i^{\,2}) \tag{5}$$

and

$$\mathrm{Tan}^{-1}\frac{b}{a} - \left|\mathrm{Tan}^{-1}(\omega T_1) + \mathrm{Tan}^{-1}(\omega T_2) + \ldots + \mathrm{Tan}^{-1}(\omega T_{n-1})\right| = -\left(\frac{1}{2} + 2n\right)\Pi \tag{6}$$

Using the substitutions:

$$\mathrm{Tan}^{-1}\frac{b}{a} = C,\ \mathrm{Tan}^{-1}\,(\omega T_i) = d_i,\ \text{and} \sum_{i=1}^{n-1} d_i = D \tag{7}$$

into (6), the following is yielded upon simplification:

$$\mathrm{Tan}\ C \cdot \mathrm{Tan}\ D = -1 \tag{8}$$

The relationship given in (8) is important because it indicates that at the limit cycle, the composite slope of the linear frequency response function, denoted by Tan D, is perpendicular to the slope of the nonlinearity, $\frac{b}{a}$ or Tan C. The derivation of the composite slope of G(s) is given separately in Appendix A. Let the $n^{\underline{th}}$ order control system be even for sake of argument. From (A-6) of Appendix A, the expression for Tan D becomes:

$$\mathrm{Tan}\ D = \frac{1}{j}\,\frac{(j\omega)\sum_i^{n-1} T_i + (j\omega)^3 \sum_i^{n-3}\sum_j^{n-2}\sum_k^{n-1} T_iT_jT_k + \cdots + (j\omega)^{n-1}\sum_i^{1}\sum_j^{2}\cdots\sum_z^{n-1} T_iT_j\cdots T_z}{1 + (j\omega)^2 \sum_i^{n-2}\sum_j^{n-1} T_iT_j + \cdots + (j\omega)^{n-2}\sum_i^{2}\sum_j^{3}\cdots\sum_y^{n-1} T_iT_j\cdots T_y} \tag{9}$$

Using (7) and (9), (8) simplifies to:

$$\mathrm{Tan}\ D = \frac{1}{j}\,\frac{\prod_1^{n-1}(1+j\omega T_i) - \prod_1^{n-1}{}_1(1-j\omega T_i)}{\prod_1^{n-1}(1+j\omega T_i) + \prod_1^{n-1}(1-j\omega T_i)} = -\frac{a}{b} \tag{10}$$

Combining (5) and (10), and after algebraic manipulations, conditions for sustained oscillations result as:

$$\prod_1^{n-1}(1+j\omega T_i) + \prod_1^{n-1}(1-j\omega T_i) = -2\,\frac{bK}{\omega} \tag{11a}$$

$$\prod_{1}^{n-1} (1+j\omega T_i) - \prod_{1}^{n-1} (1-j\omega T_i) = -\frac{2aK}{j\omega} \tag{11b}$$

The gain margin, (5), and any two of (8), (11a), and (11b) are used to determine the amplitude and frequency of oscillation at the limit cycle. It is important to point out that (11a) - (11b) give rise to a characteristic equation of order n:

$$1 + (j\omega)^2 \sum_{i}^{n-2} \sum_{j}^{n-1} T_i T_j + \cdots + (j\omega)^{n-2} \sum_{i}^{2} \sum_{j}^{3} \cdots \sum_{y}^{n-1} T_i T_j \cdots T_y = -\frac{bK}{\omega}$$

$$\omega \sum_{i}^{n-1} T_i - \omega^3 \sum_{i}^{n-3} \sum_{j}^{n-2} \sum_{k}^{n-1} T_i T_j T_k + \cdots + j^{n-2} \omega^{n-1} \sum_{i}^{1} \sum_{j}^{2} \cdots \sum_{z}^{n-1} T_i T_j \cdots T_z = +\frac{aK}{\omega} \tag{12}$$

This is intuitively true, since G(s) is of order n. Therefore, determination of limit cycle conditions reduces to the solution of (12). Note that if G(s) does not contain a pole at the origin, i.e., s=0, then the phase margin condition (6) reduces to:

$$\operatorname{Tan} D = \frac{b}{a} \tag{13}$$

In other words, the composite slope of G(s) and the slope of the nonlinearity are equal for this situation. To verify the derived conditions for sustained oscillations, a few cases are considered.

EXAMPLES

Let a fourth order open loop transfer function be

$$G(s) = \frac{K}{s(1 + sT_1)(1 + sT_2)(1 + sT_3)}$$

in the presence of a multi-valued type nonlinearity. Using (12), one obtains a set of equations:

$$\omega^2 = \left(1 + \frac{bK}{\omega}\right) / \sum_{i}^{2} \sum_{j}^{3} T_i T_j \tag{14a}$$

$$\omega = -\left(\frac{aK}{\omega}\right) \left[\left(1 + \frac{bK}{\omega}\right) \frac{\sum_{i}^{1} \sum_{j}^{2} \sum_{k}^{3} T_i T_j T_k}{\sum_{i}^{2} \sum_{j}^{3} T_i T_j} - \sum_{i}^{3} T_i\right]^{-1} \tag{14b}$$

Eliminating ω from this set and making use of the fact that at the limit cycle at least one of the roots must be on the jw axis, it yields into

$$-(K/4) \left[\sum_{i}^{3} T_i \left(\frac{\sum_{i}^{2} \sum_{j}^{3} T_i T_j}{\sum_{i}^{1} \sum_{j}^{2} \sum_{k}^{3} T_i T_j T_k} \right)^2 - \frac{\sum_{i}^{2} \sum_{j}^{3} T_i T_j}{\sum_{i}^{1} \sum_{j}^{2} \sum_{k}^{3} T_i T_j T_k} \right]^{-1} = \frac{a}{b^2} \tag{15}$$

When the plot of a/b^2 intersects the quantity on the LHS of (15), the limit cycle occurs. At this intersection, the amplitude of oscillation is found from a plot of a/b^2 and the frequency of oscillation is determined from (14). The limit cycle conditions for this control system are shown in Table 1.

SPECIAL CASES

If a pole at the origin from a fourth order control system is removed, then limit cycle conditions for a third order control system are obtained by replacing $(\frac{b}{\omega})$ for a and $(\frac{a}{\omega})$ for -b in (14a)-(14b) respectively. These conditions are shown as Case 2 in Table 1. Table 1 also shows some more special cases of third order and second order control systems with and without a pole at the origin. The results of second order and third order control systems with a pole at the origin, Cases 5 and 3, are identical to those obtained earlier by Sen [3] and Dudkikar [1].

As an illustration of Case 3 in Table 1, consider a closed loop feedback control system with

$$G(s) = \frac{3}{s(0.5s + 1)\ (s + 1)} \tag{16}$$

involving a type "L" nonlinearity described by its describing function as

$$N(E) = a(E) + j\ b(E)$$

$$\text{where } a(E) = \frac{4M}{\Pi E} \cos \theta,\ b(E) = - \frac{4M}{\Pi E} \text{Sin } \theta,$$

$$\theta = \text{Sin}^{-1} \left(\frac{c}{E}\right), \text{ and } c = M = 1. \tag{17}$$

Since $T_3 = 0$, (14a) and (14b) reduce to:

$$\omega^2 = 2a, \text{ and } \omega = \frac{3b}{a-1} \tag{18}$$

Using (18), it is found that the amplitude of oscillation at the limit cycle is about 2.713 units and correspondingly the frequency of oscillation is found to be 0.947 radians per unit period of time. This result agrees with Reference 1.

Column 3 of Table 1 is used as a basis for determination of limit cycle stability. The third order control system under Special Cases will be stable for all amplitudes where:

$$\frac{a(a-1)}{b^2} > 4.5 \text{ units}$$

is satisfied, since the roots of its characteristic equation will lie in the left half complex plane. For $\frac{a(a-1)}{b^2} < 4.5$ units, the unstable limit cycle will occur.

SUMMARY

This paper has shown the phase margin and gain margin technique used to analyze the nature of a closed loop n^{th} order nonlinear control system limit cycle. A generalized expression to compute the composite slope in terms of individual slopes is also derived.

ACKNOWLEDGMENT

The author would like to thank Mr. R. G. Nystrom of The MITRE Corporation, McLean, Virginia, for invaluable assistance in preparing this paper.

REFERENCES

1. Dudkikar, K. L, "Third Order Control System with Type L Nonlinearity," First Annual Houston Conference on Circuits and Systems, May 22, 1969.
2. Gibson, J. E., Nonlinear Automatic Control, McGraw Hill Book Company, Inc., New York, N.Y., 1963.
3. Sen, A. K., "A Simpler Stability Criterion for a Class of Nonlinear Systems, IEEE Trans. Automatic Control, August 1968.

APPENDIX A

GENERAL EXPRESSION FOR COMPOSITE SLOPE

The general formula is developed to compute the Tangent of sum of different angles expressed in terms of individual angles.

Let the different angles be $d_1, d_2, \ldots, d_n$ each denoted by:

$$d_i = \mathrm{Tan}^{-1}(a_i) \text{ and } \sum_{i=1}^{n} d_i = D \tag{A-1}$$

Also the subscripts, $j = i + 1$, $k = j + 1$, $\cdots$, $z = y + 1$. (A-2)

The problem is to find Tan D. Since

$$\mathrm{Tan}\left(\sum_{i=1}^{2} d_i\right) = (a_1 + a_2)/(1 - a_1a_2)$$

Tan $\left(\sum_{i}^{3} d_i\right)$ becomes, upon simplifying, as (A-3)

$$\mathrm{Tan}\left(\sum_{i=1}^{3} d_i\right) = \left(\sum_{i}^{3} a_i - \sum_{i}^{1}\sum_{j}^{2}\sum_{k}^{3} a_ia_ja_k\right)/\left(1 - \sum_{i}^{2}\sum_{j}^{3} a_ia_j\right) \tag{A-4}$$

Using similar reasoning, one writes:

$$\mathrm{Tan}\left(\sum_{i=1}^{4} d_i\right) = \left(\sum_{i}^{4} a_i - \sum_{i}^{2}\sum_{j}^{3}\sum_{k}^{4} a_ia_ja_k\right)/\left(1 - \sum_{i}^{3}\sum_{j}^{4} a_ia_j + \sum_{i}^{1}\sum_{j}^{2}\sum_{k}^{3}\sum_{l}^{4} a_ia_ja_ka_l\right) \tag{A-5}$$

By repeating this process further up to n angles, the general formula is obtained and the final result is

$$\mathrm{Tan}\left(\sum_{i=1}^{n} d_i\right) = \frac{j\sum_{i}^{n} a_i + j^3 \sum_{i}^{n-2}\sum_{j}^{n-1}\sum_{k}^{n} a_ia_ja_k + \cdots + j^{n-1}\sum_{i}^{2}\sum_{j}^{3}\cdots\sum_{y}^{n} a_ia_j\cdots a_y}{j\left\{1 + j^2\sum_{i}^{n-1}\sum_{j}^{n} a_ia_j + \cdots + j^n \sum_{i}^{1}\sum_{j}^{2}\cdots\sum_{z}^{n} a_ia_j\cdots a_z\right\}} \tag{A-6}$$

where n is even. If n is odd, Tan D is obtained by only replacing the last term in numerator by the last term in denominator and vice versa, and by rearranging all other missing terms in both numerator and denominator.

TABLE 1
EXPRESSIONS FOR SUSTAINED OSCILLATIONS FOR VARIOUS G(s)

	G (s)	Equation 11a, ω^2 equals*	Equation 11b, ω equals	Plot Intersection
1.	$\frac{K}{s(1+sT_1)(1+sT_2)(1+sT_3)}$	$\frac{1+\frac{bK}{\omega}}{\sum_i^2\sum_j^3 T_iT_j}$	$\frac{-(aK/\omega)}{\left(1+\frac{bK}{\omega}\right)\frac{T_1T_2T_3}{\sum_i^2\sum_j^3 T_iT_j} - \sum_i^3 T_i}$	
2.	$\frac{K}{(1+sT_1)(1+sT_2)(1+sT_3)}$	$\frac{1+aK}{\sum_i^2\sum_j^3 T_1T_j}$	$\frac{bK}{(1+aK)\frac{T_1T_2T_3}{\sum_i^2\sum_j^3 T_iT_j} - \sum_i^3 T_i}$	
3.	$\frac{K}{s(1+sT_1)(1+sT_2)}$	$\frac{1+\frac{bK}{\omega}}{T_1T_2}$	$\frac{aK/\omega}{\sum_i^2 T_i}$	$K\sum_i^2 T_i \doteq \frac{a}{b^2}\left\{\frac{aKT_1T_2}{\sum_i^2 T_i} - 1\right\}^2$
4.	$\frac{K}{(1+sT_1)(1+sT_2)}$	$\frac{1+aK}{T_1T_2}$	$-\frac{bK}{\sum_i^2 T_i}$	$\frac{(T_1+T_2)^2}{T_1T_2} = \frac{b^2K^2}{1+aK}$
5.	$\frac{K}{s(1+sT_1)}$	b^2K^2	$\frac{aK/\omega}{T_1}$	$KT_1 = a/b^2$

*Subscripts $j = i+1$, and i is a positive integer.

AN OPERATIONAL EXPERIMENT FOR THE MARKETING OF SCIENTIFIC AND TECHNICAL INFORMATION

DONALD J. HILLMAN
Lehigh University
Bethlehem, Pennsylvania 18015

ABSTRACT

A new NSF-supported program of research is described, in which a systems approach is applied to the decision problems of marketing scientific and technical information within the environment of producers, vendors, processors, distributors, and users. The paper places primary emphasis on conceptual models of the marketing process.

1. THE PSEF EXPERIMENT

The Pennsylvania Science and Engineering Foundation (PSEF) is conducting an experiment with the support of the National Science Foundation[1] to test the marketing of new scientific and technical information products with the objective of increasing the economic user base for, and improving the receptivity to, scientific and technical information, especially among professional user groups engaged in industrial research and development. In this experiment, the problem of marketing a technical information service is regarded as a managerial decision problem in which the overall effectiveness of the information service organization must be optimized over the frequently conflicting objectives of such functional units as data base update facilities, communication networks, data base vendors, and sales activities. This systems approach to data base sales forms the basis of a general economic study of cost-benefit trade-offs, and is being followed by a multidisciplinary team. The Pennsylvania Science and Engineering Foundation coordinates the activities of the Bureau of Planning and Research and the Bureau of Economic Development within the Pennsylvania Department of Commerce in providing economic data on income, prices, orders, inventories, employment and other industrial data. The Pennsylvania Technical Assistance Program (PENNTAP) provides a technology transfer capability *via* a statewide network of access points to the information service. Of great importance in this respect is PENNTAP's role in defining *usable forms* of information at the small and relatively unsophisticated user interface. The Temple University Bureau of Business and Government Services provides the focus and overall direction of the project, and coordinates the role of the Temple University Marketing Department in the activity. Finally, Meta-Science, Inc., provides marketing research and conducts test marketing, while the LEADERMART information system provides the technical basis for the complete marketing experiment.

The unique character of the PSEF experiment is reflected not only in the nature of the multidisciplinary team that has been assembled, but also in the novelty of the experimental design, as indicated below.

2. MARKETING ANALYSIS OF DOCUMENT TRANSFER SYSTEMS

The PSEF experiment is concerned with the operation of document transfer systems, where "document" is liberally interpreted to include bibliographic data, abstracts, etc, as well as primary journal articles. In other words, the domain of the experiment comprises those current-generation information retrieval systems that are bibliographically oriented, i.e. do not perform data or fact retrieval. In addition, the experiment is focused upon the operations of information service or dissemination centers, so that only the functions of document acquisition,identification, and delivery are the subject of study. The remaining functions of a document transfer system, viz., those associated with the preparation, editing, and publication of documents at the production end, and assimilation of documents at the user end, are outside of the scope of the PSEF experiment. Thus, in the terminology of King and Bryant,[2] the experiment is concerned with the proprietary aspects of document transfer systems, i.e. acquisition, identification, and delivery, in relation to users.

This delimitation of scope has certain important advantages. Most significantly, we are now in a position to treat document transfer systems within a marketing environment. Specifically, we regard the users of document transfer systems to be outside of the proprietary scope of system activity, so that the marketing of documentary products and services can be studied in terms of modern marketing techniques. It should be carefully noted, however, that the fact that the user is not part of the proprietary system in no way implies that the user is not part of the transfer system. Of course he is, but he is not under the management control of the information service organization.

This distinction between the proprietary and non-proprietary aspects of document transfer systems allows us now to sharpen on statement of objectives. It is appropriate to say that our marketing objective is to increase user satisfaction with the products of the document transfer system, thus stimulating increased purchases of those products. This differs from older marketing concepts, which were oriented more toward products than to users, with a corresponding emphasis on volume/cost considerations as the primary means of economic viability.

With respect to users themselves, we have decided to confine our attention to the information specialists of business and industrial organizations. This is a deliberate choice which has been amply corroborated by the results of the marketing research phase of the PSEF experiment. As was expected, the information specialist is a ubiquitous intermediary between the ultimate or end user of scientific and technical information and the identification subsystem of the document transfer system. In this connection, we are interested in establishing a continuum of institutional users, from the very large to the quite small, and evaluating both the effectiveness of different marketing techniques and the benefits of information products as perceived by information specialists.

With regard to institutions, whether large, medium, or small, our concern is to determine how marketing techniques can lead to wider product acceptance within existing markets for information services. We have chosen to attack this problem by way of scientific and technical information innovations, i.e. specially designed and customized user-oriented packages featuring different data base blends and document identification procedures. The LEADERMART system provides these user-oriented packages and its extensive text-processing software, on-line file structures, and natural language retrieval capabilities have proved to be ideally suited to the purposes of the PSEF experiment.

Another equally important class of individuals in the experiment is the widely dispersed and previously unstudied class of non-users of scientific and technical information. The Pennsylvania Technical Assistance Program (PENNTAP) provides us with a direct entree to this most interesting melange of real (sometimes critical) requirements, answer-seeking behavior, a renaissance of the county agent concept, and diffuse information sources that together make up the phenomenon of technology transfer. Here, the end users of scientific and technical information rarely resemble their counterparts in even quite small industrial laboratories and other organizations. Their needs have been summarized as "facts, not folios", i.e. they need solutions to problems rather than bibliographies. In the PSEF experiment we have decided to study the needs of such users once again by way of information specialists, in this case the PENNTAP specialists at the Pennsylvania State University. This choice not only preserves the unity of our experimental design, but also allows us to study at least two very different types of behaviors, viz., that of an information specialist in his internal company environment, and that of a PENNTAP information specialist in translating his own use of literature sources into specific answer-providing solutions.

In deciding to focus attention on information specialists in our marketing study, we are recognizing a very definite status quo in the information consumer market. Quite literally, user satisfaction with information products and services means, for the most part, satisfaction of information specialists with those products and services. It is, however, of obvious and crucial importance to consider what happens to those products after they leave the information specialist. In this connection, we have built into our experiment a study of the relationships between information specialists and those they serve in industrial organizations, on the one hand, and information specialists at PENNTAP and their clients, on the other. This part of the study is of vital importance in evaluating the impact of information services on the end or ultimate user, for it is only here that we can measure the extent to which information is assimilated.

In summary, the PSEF marketing experiment is directed at the information specialist in the present industrial market to measure the effectiveness of certain techniques in marketing new and innovative information products with the aim of increasing market share. It is also directed at the information specialist in his role of transforming information to practical solutions in a technology transfer environment. Here, the aim is to create new markets rather than expand existing ones. The information specialist is the focus of these twin activities, and relative to whatever level of performance he specifies, our task in the marketing experiment is to measure costs and benefits. Furthermore, since the motivations of institutions and institutional users are affected by the price of services as well as their quality, we are studying the problem of an optimal pricing strategy. This is intimately linked to the problem of quantifying benefits.

3. CONCEPTUAL FRAMEWORK OF THE MARKETING ANALYSIS

We consider that the decision-maker in the marketing of scientific and technical information is the information service manager, and that his role is to evaluate and compare the various alternative actions that are open to him. These controllable factors interact with the uncontrollable factors associated with the non-proprietary parts of document transfer systems such as vendor prices, on the one hand, and consumer demand and competition, on the other. We introduce the notion of an outcome array to analyze the interaction of controllables and uncontrollables.

We next introduce the concept of utility to allow us to translate outcome arrays to payoff arrays, each element of which is a single utility number that includes all the pertinent aspects of the information service organization's objectives and their degree of satisfaction. These are then analyzed in terms of risks and uncertainties. Several criteria for choice under uncertainty have been proposed, including the maximin, minimax, maximax, Hurwicz alpha, Laplace, and regret criteria, etc. This abundance of criteria is an indication of the conceptual difficulties of marketing decision-making, difficulties that militate against any anayltic solution that is developed for the (necessarily) ill-defined likelihood situations encountered in the marketing of document transfer system services. A further complicating factor is that for the information service organization utility is defined in terms of monetary value, whereas for the information specialist in the company utility is not nearly as easily quantifiable in dollar terms, although management in making dollar allocation for information services may disagree. In other words, the "bottom up" marketing of information products may suffer from inconsistencies of purpose within the customer organization, posing corresponding problems of measuring the benefits of scientific and technical information. This situation may even have led to the use of monetary value as a proxy for utility because so little is known about the latter notion so far as information is concerned. We hope to make some progress toward a better understanding of the utility of information products as a result of our study of the perceived benefits of the innovations involved in the PSEF experiment.

4. MARKETING MODELS

In the analysis of marketing decision problems, there is a clear and continuing need for some kind of conceptual apparatus that facilitates drawing the kind of conclusion usually arrived at through experimentation. Models fulfill this role, where by "model" we mean a representation of a document transfer system which is used to predict the effect of changes in certain aspects of the system on the performance of the system. Morris[3] has suggested the following steps for constructing models:

(a) Identify and formulate the manager's decision in writing.

(b) Identify the constants, parameters, and variables involved. Define them verbally and then introduce symbols to represent each one.

(c) Select the variables that appear to be most influential so that the model may be kept as simple as possible. Distinguish between those that are controllable by the manager and those that are not.

(d) State verbal relationships among the variables based upon known principles, specially gathered data, intuition, and reflection. Make assumptions or predictions concerning the behavior of the noncontrollable variables.

(e) Construct the model by combining all relationships into a system of symbolic relationships.

(f) Perform symbolic manipulations.

(g) Derive solutions from the model.

(h) Test the model by making predictions from it and checking against real-world data.

(i) Revise the model as necessary.

This general procedure of model construction has been adopted in the PSEF experiment, and is commented on briefly below. Here, it is sufficient to say that the model we have constructed will be used to generate the predicted outcomes associated with strategies, competitive actions, and uncontrollable factors with respect to the outcome array in our conceptual framework. In other words, our model is a device for predicting the performance level associated with each outcome. As for measures of performance, it must be constantly borne in mind that whatever measure or measures we choose are dictated by the objectives of the experiment. For example, one of our objectives is to determine the performance of scientific and technical information innovations in the existing market of medium to large industrial organizations. Here, we are concerned with revenue, market penetration, and image, so that the descriptors of "pure" outcomes might be "dollars of revenue", "market share", and some verbal description of the image level. Our model must therefore permit the prediction of these performance measures, and a number of outcome descriptors, each with its own dimension, is the best choice for measuring performance.

Another objective of the PSEF experiment is to measure the perceived benefit of an information service to users, where "users" is subject to the differences of interpretation mentioned earlier. In this case, our procedure is to express other aspects of the experiment's objectives, such as revenue and market share, as constraints on the basic objective. Thus, we plan to maximize user benefits subject to the constraints that given revenue, market share, and image levels be maintained. In other words, our model must be sufficiently flexible to incorporate a number of experiments with different objectives. What our model is, and what steps we went through in its construction, are briefly described below.

5. MARKETING RESEARCH

A series of workshops, personal interviews, and telephone interviews has been completed in order to gather, record, analyze, and report salient facts relating to the transfer of scientific and technical information from data base vendors through information service centers to information departments of small to large scale industrial organizations. This has resulted in a clear identification of user preferences for information products, and an equally clear indication of interest in certain specialized information packages not now being provided.

6. PRODUCT SEARCH

On the basis of our marketing research, we have been able to identify the key factors and parameters involved in the marketing decision-making process being studied in the PSEF experiment. In the product search phase, we are refining this intelligence into a description of controllable and uncontrollable variables, together with an explication of the relationships holding between them. Among controllable variables are available scientific information services and products, available data bases, and available computer resources. Among uncontrollable variables are user demand and certain prices and costs. The Pennsylvania Department of Commerce and its various Bureaus are an excellent source of the type of market data required for product search and planning. Each new product idea will also be rated in terms of its marketability, durability, and productive ability in order to calculate utility measures. This will complete the delineation of innovations that appear to be promising enough to warrant further analysis. By this time, we will have completed the first four steps of our model construction procedure.

7. PRODUCT EVALUATION

This task represents the basic decision problem of information product planning, and will be approached by an economic analysis involving fixed and variable costs. We will combine all relationships identified so far between controllable and uncontrollable variables into a system of symbolic relationships covering advertising and promotional costs, delivery schedules, networking considerations, data base usage, customized information packages, and data base processing operations. We shall apply mathematical programming techniques to study interlocking decisions on the production and distribution of information innovations, perform symbolic manipulations on our model, and derive solutions in the form of the best program for each alternative. This will take us through step 7 of model construction.

8. TEST MARKETING AND BENEFIT EVALUATION

In order to test the model we have developed, we shall design a testing program for each of the innovations described above, select appropriate performance variables, such as market share for services to existing markets, and benefits to users for new markets (either end users in industry, or previous non-users), and evaluate performance in terms of multiple-dimension outcome descriptors. Of importance to the service center is the development of an optimal pricing strategy, and the institution of good performance controls. That is, we must be able to take corrective actions which will prevent long-term harm to the products being test-marketed, and enable the service organization to strive toward a good combination of controllables. In addition, we wish to measure the benefits of the innovations for existing users and new users, and recommend a procedure to service center managers that will maximize user benefits subject to the constraints of costs. Finally, we shall forecast the expanded market for our innovations with the help of a penetration model, and compare with actual marketplace performance.

It is apparent from the above that the conceptual apparatus and modeling techniques used in the PSEF experiment are very amenable to computerization. This, of course, was a prime factor in their selection. In addition, there are important implications in the PSEF study for the analysis and design of information dissemination networks, especially with respect to nonconventional or innovative information packages. The PSEF experiment is therefore a significant component of new investigations into knowledge transfer and delivery systems, in which documents are no longer the sole vehicles of knowledge.

REFERENCES

1. Grant GN-42271, 4/1/74 - 3/31/76.

2. Donald W. King and Edward C. Bryant, The Evaluation of Information Services and Products, Information Resources Press, Washington, D.C., 1970.

3. W. Morris, "On the Art of Modeling", Management Science, August, 1967.

THE USE OF HIERARCHIC CLUSTERING IN FORMING A SIMILARITY MEASURE

SCOTT E. PREECE
IIT Research Institute
Chicago, Illinois

ABSTRACT

Work on clustering methods in information retrieval at the IIT Research Institute is described. Problems with conventional similarity measures are discussed, and a potentially useful new similarity-measuring method is defined.

CLUSTERING AT IIT RESEARCH INSTITUTE

Over the last year or so the Information Sciences section of the IIT Research Institute has been experimenting with the use of clustering as a post-retrieval mechanism for use with certain SDI profiles that are inherently low in precision or generate results scattered over several areas. Our thought has been that clustering could reduce the problem of selecting the relevant citations from search output to one of scanning a few representative citations from each cluster. We are not particularly worried by the fact that the clustering time is on the order of N^2, since we will be dealing with relatively small values of N.

In the course of this effort we have been studying the existing literature for guidelines and also performing pilot experiments to test methods and observe the outcomes of various choices. At present we are dealing solely with very small tests on relatively uncontrolled data, simply exploring various aspects of the problem before committing a major effort to building an efficient clustering program.

The specific problems we mean to handle using clustering result from some unusual profiles. One variety of problem profile retrieves huge numbers of irrelevant documents because some relevant documents are retrievable only by a few very common words. One of the Cranfield queries (190 (24), reported in (1), p. 147) exhibits this behavior -- of its seven relevant documents, two are retrieved only by the word 'flow', the most common word in the Cranfield corpus. Clustering may not always be able to isolate the relevant documents; if a clear pattern were available for such a query's relevant documents, the irrelevant documents could be eliminated. We think it may help in some cases, however.

We feel clustering will also be useful for some search profiles which cover several related areas of interest. Thus a profile on operating systems, time-sharing systems, and compilers would be well-served by a clustering algorithm which could separate documents into the separate areas.

A third use is in processing the results of retrospective searches. In some cases an attempt to write a high-precision retrospective search may result in recall failure for reasons that could not be predicted. Use of a clustering method to separate out the good results from a low-precision/high-recall search could help avoid the cost of expensive reruns.

We would also like to use the clustering ability as a tool in designing search queries. We have designed an interactive program to make various tests available to the profiler -- such as inverted searches of a small sample data base and access to term frequencies for a volume of Chemical Abstracts Condensates -- and we would like to include in that system the ability to cluster terms or documents on request. (10) If, for instance, the profiler already had obtained a group of citations known to be relevant, and another group known to be similar but not relevant, he could cluster the terms in them to find an effective search strategy.

Our research in building towards these capabilities, and other planned research in clustering, has thus far taken the form of relatively small experiments intended more as design guides than as major research. Typically they consist of performing a clustering on a given set of documents, using a given method, and comparing the results to previous results from other methods. The funding committed to this work, the staff time available for it, and its low priority within our schedule have not permitted larger or more detailed investigation. "Comparison of results" has generally consisted of simple common sense comparison, rather than exhaustive statistical comparison of techniques. Our bibliographic research, however, has been more comprehensive. The material covered in this paper is related to such a pilot experiment, dealing with similarity measures. We do not have solid statistical results to report. We have, instead, a method to which we have not been able to find any previous reference, which seems promising to us.

We have concentrated thus far on hierarchic clustering, using an agglomerative algorithm like that of Lance and Williams (11). We felt that an hierarchic scheme was possibly better-defined than the various graph-theoretic methods, in that it leads to a clear result in the form of a dendrogram without requiring the use of a threshold parameter. This was a somewhat arbitrary choice; the literature did not indicate any clear preference among the various methods.

Our study of the literature on similarity measures, on the other hand, showed a clear consensus. Virtually everyone used either the cosine function (especially those using the SMART system), or the Tanimoto measure (especially the British researchers). Our initial efforts were made using the simple measure:

$$D(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

where $|A \cap B|$, $|A \cup B|$ are the number of shared terms and the total number of unique terms associated with the two documents A and B. For binary attributes, such as term presence or absence, this is equivalent to the Jaccard and unweighted Tanimoto measures. We compute this distance in a way that will permit the use of weights in future experiments, but for now it is left unweighted.

PROBLEMS WITH CONVENTIONAL MEASURES

Initial analysis of experiments using this measure showed few surprises. We were experimenting variously with title words, keywords, and search terms and verifying hand-computed results to check program validity. We observed, however, that the small size of our samples (25-100 documents) revealed some failures of the similarity measure. Inspection of the results immediately pointed to the causes. One common problem involved our own SDI processing: we remove keywords which also appear in the title of the same document. Thus, when we were only clustering keywords, some terms were not appearing in documents they belonged in, and their similarity to other terms was being missed. A second problem, in tests where both title words and keywords were clustered, was the non-clustering of different forms of a single word. "Computers" and "computer", for instance, showed up with no similarity indicated. A third problem resulted from the differences in indexing vocabularies among the sources of our data bases. The term from Engineering Index which exactly corresponds to a given, different term in Condensates might show up with no similarity indicated because it did not share any documents with the CA term.

For some time we considered dropping the clustering concept because we were not willing to get involved in the problems of recognizing stems and affixes and reconciling vocabularies. The use of variant forms and split vocabularies simply resulted in too sparse a coincidence matrix. We felt we had to abandon the conventional distance function.

A further search of the literature revealed that practically every author's discussion of similarity measures pointed out that a great array of such measures were available, then proceeded to use Tanimoto or cosine for his own work. One in particular stated "the difference in retrieval performance achieved by different measures is insignificant, ... one would expect this since most measures incorporate the same information." (6) This was, of course, exactly our difficulty. Every measure we found discussed was a function of two documents' term vectors (or, alternatively, of two terms' document vectors). It seemed to us that there was no convenient way to connect the terms which were being left incorrectly unrelated using these methods.

AN ALTERNATIVE APPROACH

One statement in another paper stood out as a possible alternative: "... the output of any hierarchic cluster method is... equivalent to an ultrametric dissimilarity coefficient."(3) This seemed, on many counts a preferable approach -- that the similarity between documents should not be related strictly to their co-occurring terms, but rather to the similarities among the terms they contain. The argument sounds even more convincing for term clustering -- the similarity between two terms should be based on the similarties among the documents which contain them. Thus we avoid the rationale that all measures are based on the same information by dramatically increasing the information basis of our similarity function's values.

In order to examine the change in the number of values on which each similarity measurement is based we must first define our measure. In fact we will define a whole family of measures, all basing their values on the similarity of the orthogonal elements of the system. The six measures vary in complexity of computation and amount of information extracted during the first pass. We will define each, give some general comments, then comment on each in turn, as they have some diverse properties. Each definition will be given from the point of view of terms being clustered. Thus the quantity defined is D(A,B), the distance between two terms A and B based on the distances among the documents which contain them. The document term coincidence matrix is DT, where DT(I,A) is 1 if term A is present in document I and otherwise 0. The inter-document distances, obtained from Tanimoto's measure, are in array DD, where DD(I,J) is the distance between documents I and J. For measures 4-6 a dendrogram is available from the first-stage clustering.

1. *Mean distance*. This measure returns the arithmetic mean of the distances between all pairs of documents (I,J) such that document I contains term A and J contains B.

2. *Percentage nearby*. This measure returns the percentage of distances between pairs of documents (I,J), such that document I contains term A and J contains B, which are greater than a threshold ϕ, where ϕ is a system parameter.

3. *Distance at cutoff*. This measure sorts all the distances used in measures 1 and 2 into ascending order and returns the distance below which $\phi\%$ of them lie, where ϕ is a system parameter.

4. *Percentage clustered nearby*. This measure returns the percentage of distances between document pairs (I,J) such that document I contains term A and J contains B (where the distance between two documents is the height in the dendrogram at which those two documents first are in the same cluster), that lie above the threshold ϕ, where ϕ is a system parameter.

5. *Clustered distance at cutoff*. This measure is like 3, except that distance is measured as for 4, using a complete clustering.

6. *Mean clustered distance*. This measure is like 1, except

that distance is defined as for 4, using a complete clustering.

For reasons we will discuss later, we must impose one further restraint: If A = B then we must force the result value to zero. This amounts to accepting in the calculation only document pairs (I,J) such that I = J.

The idea of using as a distance the level at which two documents are first clustered together is well established (3,4,6,8). The rationale for defining six measures is to provide for varying levels of computational complexity and their varying results. We are currently in the process of comparing all these methods. This paper is based on extremely limited results and is in no way intended as a definitive analysis of these measures.

In order to be accepted as distance measures, our six functions must meet three tests:

1) $D(A,B) \geqslant 0$ for all terms A,B
2) $D(A,A) = 0$ for all terms A
3) $D(A,B) = D(B,A)$ for all terms A,B

It is easy to see that in all cases condition three holds, since in calculating the measure exactly the same document pairs will be found, whichever term is first, and the distances between those documents satisfy condition three by definition, since cases 1-3 involve a conventional distance measure and cases 4-6 involve the result of an hierarchic clustering, which is proved by Jardine and Sibson to be a distance measure(4,8). The first condition holds for 1, 3, 4 and 6 because they are distances and for 2 and 5 because they are percentages. The second test would have been a problem except for our special constraint. The fact that the measures have to consider all pairs of documents such that one contains the first term and the other the second, or in this case all pairs of documents containing A, means that they would not generally return zero for D(A,A) except for our special condition which forces them to zero. Given our rule, however, the second test is obviously met. In practice we will never take D(A,A) anyway.

COMMENTS ON IMPLEMENTATION

Our first pass uses the Tanimoto measure and a Group Average agglomerative algorithm to cluster documents. Term-term distances, where non-zero, are then passed to a second program for term clustering. In both cases only one side of the symmetric similarity matrix is stored (those values DD(I,J) where J > I or of D(A,B) where B > A). The term clustering, after reading the term-term distance matrix, uses the same algorithm as the document clustering did. For our present uses this algorithm is acceptable -- we can handle up to a 500 x 500 distance matrix on the computer we use, and we are only planning on clustering rather smaller sets. The program has not been optimized yet, nor have we made any attempt to find a faster algorithm; at this point we are interested only in results and techniques, not efficiency.

The cost of our six similarity measures is significantly higher than the cost of the conventional measures. At present the simplest of them adds about 50% to the cost of a clustering using somewhat smaller sets of documents than we would be using in production. The more complicated ones are more expensive than the entire second pass clustering. This is a definite drawback. For measures 4-6 we are adding a complete second clustering, plus the cost of gathering the values for the similarities for each distance used in the second pass (descriptions of the algorithms are given below). The document:term coincidence matrix is still built only once, but there is also some added file overhead. Using our present programs the cost of clustering 50 documents with about 8 terms each, using the conventional measure, is about 20 CPU seconds. For 100 documents we would expect it to rise to about 65 seconds. Before production use we expect that to fall, through program improvement, to about 25 seconds for 100 documents with 15 terms each, which we would expect to be typical of our uses. The second stage clustering would then take around 20 seconds. These figures reflect our consideration of the programming methods we used and a reasonable estimate of the improvements readily available.

With a heterogeneous collection it should be possible to improve on N^2 performance by the simple expedient of partitioning the collection into groups with no interconnections. This partitioning can be done fairly cheaply, in fact, by simple monitoring as the documents are read into the system. In our use, however, the collections being clustered are likely to be quite homogeneous, and in most cases every document will have some connection with every other document. Our limited tests have exhibited exactly this behavior. Every document was tied at some distance <1 to about 90% of the other documents. Every document was tied to at least one document that was tied to each other document. Clearly no partitioning is possible.

We did find that a large part of the document term matrix could be collapsed by removing terms which occur only once. These would be replaced by dummies for association calculation. In calculating document associations, the specific terms are unnecessary, since they cannot add to the association. In the second pass of a term clustering, however, they cannot be replaced. By the nature of our new measures, two terms which occur only once may have some similarity because of the similarity of the documents containing them.

As to the effectiveness of these measures, we can only say that they seem to behave as we had expected. Term pairs we felt should have been linked now are appearing with stronger associations than they had under conventional clustering. Two examples of this improvement follow:

	CONVENTIONAL	MEAN DISTANCE	PERCENTAGE NEARBY
A.	1.0	.65	0
B.	1.0	.43	.06

These are the distances (0 = identical, 1 = no similarity) between the terms 'ON-LINE' and 'INTERACTIVE' (A) and 'ASSEMBLERS' and 'TRANSLATORS' (B). In both cases the similarity-based measure gave significantly better results. The values under percentage nearby both reflect the threshold effect. Most of the clustering is not significantly changed from the conventional method. Several side effects, tracing back to the specifics of the clustering methods, are mentioned below with the particular methods involved.

One problem, if it is a problem, is common to the methods using percentage threshold value. This is that documents closer than the threshold distance are treated as being at zero distance. This leads to some unnatural zero distances. More precisely, if two terms occur only in a set of documents which are all within ϕ distance of each other, they will have a distance of zero returned, which is not necessarily the desired result. The mean distance methods, and the distance at cutoff methods will not have exactly this problem, but the latter will return a false zero where the requisite proportion of the values are zero, and the rest are not. The effect of these false zeros may not actually be a serious drawback. Many authors have used, or considered the desirability of using, thresholds in setting up clusters. If the anomolous behavior is known, the results may still be acceptable.

The general difference between the first three methods and those basing their values on a complete clustering is that the complete clustering involves the entire collection in the calculation of each value. That is, the distance between two documents will be affected by the entire collection's inter-document similarities. The magnitude of this effect will vary with the specific clustering method used. A space-contracting method, such as nearest neighbor (single-link), and a space-expanding method, such as farthest neighbor, will have very different effects on the resulting measure. Group average is thought to be roughly space-conserving, but some effect may be observable nevertheless.

The cluster-based methods determine each value using all $n^2/2-n$ distances in the distance matrix plus the manipulation involved in forming clusters. The similarity-based measure, on the other hand, is based on the term vectors of all and only the documents which contain the two terms. If each document contains an average of t terms, then each term is contained in an average of tn/m documents, where n is the total number of documents and m the total number of terms in the system. Each term:term distance will then be based on the term vectors associated with $2t - (t/m)^2$ documents or $2tm - t^2/m$ bits of data. This is compared to the 2n bits examined by a conventional **measure**.

Descriptions of the algorithms for the calculation of each measure are given below based on internal data consisting of the document:document distance matrix only (all are given in terms of calculating a term:term distance). This seemed the most appropriate primary form for the algorithms. Calculations would be faster and simpler, if a list were generated for each term of all documents containing that term. This would eliminate the need to scan the coincidence matrix to locate the necessary document pairs during computation, and could be done during read-in at very little cost.

Similarly, the internal form of the dendrogram resulting from the first stage document clustering is a data node for each term consisting of the number assigned to the cluster resulting from the combination of the term with another term or cluster and the distance at which the combination was made. Similar pairs are included for each of the clusters so formed. This is a reasonable form, but calculation would be simplified greatly if the document:document distance matrix were refilled, after the clustering, with the new values (namely, for each document pair, the distance at which they are first part of a single cluster).

1) The mean-distance measure is the simplest to compute. Once the document-document distances have been calculated the distance between two terms is calculated by using the coincidence matrix to locate the appropriate pairs of documents, summing their distances, and dividing by the number of pairs. The specific work involved is the locating of the items in the coincidence matrix. Using the same definitions given just above, this will mean examining n bits for each of tn/m ones in the row associated with the first item.

2) The percentage-nearby measure is no more difficult to calculate. The same pairs of documents needed for 1 are again examined, but this time a count is kept of the number which fall above the threshold and the result count divided by the number of pairs. The effort involved is the same as for 1, but the results are rather different. The threshold effect is quite noticeable, showing a signficantly increased number of zeros. Whether this is acceptable or not is unclear; the zeros correctly indicate a strong relationship, but a zero is misleading.

3) The distance-at-cutoff measure is more difficult to implement. In order to locate the ϕ percentile cutoff level, the values must be sorted. Thus the work involved in calculating a value is that of the previous two measures plus, if there are v values located by that procedure, order (v log v) effort to sort the list of values and the counting of $v \cdot \frac{\phi}{100}$ **values** to locate the cutoff value. Given the problem of the threshold-induced zeros, this measure probably is not worth the effort involved in its calculation. On the other hand, it seems logically at least as reasonable as 2.

4) The percent-clustered-nearby measure is calculated, once the clustering is done, in the same fashion as 2 above, but for each document pair obtained, a distance must be

extracted from the dendrogram. To do this two pointers are created, initially pointing to the two documents. A loop is then executed, advancing the pointer which points to the term or cluster whose level at clustering is lower so that it points to the result cluster. The loop terminates when both pointers point to the same cluster. If both pointers point to nodes with the same level, the one which initially pointed to the first document is advanced. When the pointers match, the level from the last-previous node is returned as the distance between the documents. The average distance climbed would be $(\log_2 n)/2$ nodes, and both pointers would advance that far, for a total of $\log_2 n$ advances, for a balanced dendrogram. Actual dendrograms will not be balanced, and the distance climbed will vary accordingly. The total amount of climbing is not predictable, since it will vary with the depth of the specific document pair s whose distances must be determined, and the number of times each is required.

The threshold-induced zeros that occur with measures 2 and 3 will also occur with measures 4 and 5. In both cases the zero distances will not be affected by clustering, under the algorithm we are using, because documents at zero distance will be clustered together. The first stage similarity measure returns zero distance only where two documents' terms match exactly, hence all such documents will be clustered together at distance zero.

5) The clustered-distance-at-cutoff measure requires essentially the same effort as 3, except for the increased cost of finding document-document distances, and the cost of the clustering.

6) The mean-clustered-distance measure requires the same effort as 1, except for the increased cost of locating document-document distances and the cost of clustering.

CONCLUSION

Our preliminary results show that measure 1 may be a significant improvement over conventional distance measures. The other measures have not shown any marked improvement over measure 1, so far; some specific term pairs have been better-handled, others not as well. For the data we have tried so far, the clustered measures have not performed well with low values of ϕ, but have returned relatively high distances and insufficient definition. It seem likely that measure 1 will be the measure used for our production system. Over the next few months we will experiment with several sets of data, running searches using the results to produce recall and precision values for the various term clusterings, to produce more definitive results.

We feel that the idea we have discussed here, of using information on document similarity to determine term similarity or vice versa, is workable. Whether these specific measures are the proper approach or not remains unclear pending further study, but our preliminary study does indicate that they represent an amelioration of some problems we have observed with conventional measures, and therefore, possibly a step in the right direction.

REFERENCES

1) K. Sparck Jones: Automatic Keyword Classification for Information Retrieval. Butterworths, London (1971).

2) K. Sparck Jones: Collection properties influencing automatic term classification performance. Inform. Stor. Retr. 1973, 9, 499-513.

3) N. Jardine, R. Sibson: Choice of methods for automatic classification. Comp. J. 14, 404-406.

4) N. Jardine, C. J. van Rijsbergen: The use of hierarchic clustering in information retrieval. Inform. Stor. Ret. 1971, 7, 217-240.

5) J. Minker, G. A. Wilson, B. H. Zimmerman: An evaluation of query expansion by the addition of clustered terms for a document retrieval system. Inform. Stor. Ret. 1972, 8, 329-348.

6) C. J. van Rijsbergen: Further experiments with hierarchic clustering in document retrieval. Inform. Stor. Ret. 1974, 10, 1-14.

7) G. Salton: Automatic information organization and retrieval. McGraw-Hill, New York (1968).

8) N. Jardine, R. Sibson: Mathematical Taxonomy. Wiley, New York (1971).

9) R. R. Sokal, P. H. A. Sneath: Principles of numerical taxonomy. Freeman, San Francisco (1963).

10) S. E. Preece: Design for a modular query pre-processor system. Paper to be delivered at ASIS 1974 National Convention, October, 1974.

11) G. N. Lance, W. T. Williams: A general theory of classificatory sorting strategies; I. Hierarchic strategies. Comp. J., 9, 373-380.

AUTOMATIC CLASSIFICATION OF SCIENTIFIC LITERATURE USING CO-CITATION CLUSTERING

HENRY G. SMALL
Institute for Scientific Information, Philadelphia, Pa.

BELVER C. GRIFFITH
Drexel University, Philadelphia, Pa.

ABSTRACT

A computer-based system for automatically classifying the scientific literature is described. The unique feature of this system is that its structure is completely determined by citation patterns in the Science Citation Index (SCI). The system creates clusters of highly cited documents by co-citation and then uses these clusters to classify the citing documents. The system has been tested on the 1973 SCI data base. About 1,600 clusters were formed which were capable of classifying about one-quarter (100,000) of the source documents in the SCI. Classification headings were derived by title word frequency analysis of the citing documents in each cluster.

INTRODUCTION

In this paper we will discuss a research effort, still underway, which attempts to exploit the "natural" structure of scientific literature in the design of a retrieval system. The impetus for this work, in fact, did not come from information science, but rather from an attempt to develop a means for monitoring changes in the structure of science. We hope that our efforts will mark a convergence of these two disciplines.

Science, when viewed as a set of social and intellectual processes, is essentially self-organizing, that is, it evolves its own structure internally. It is our task to discover what this structure is and explore it systematically, rather than impose a structure upon it, such as a classification scheme. We believe that the optimum organization of the scientific literature for retrieval will correspond as closely as possible to the way that literature is generated by the scientists. Perhaps the best indicators of this mode of self-organization are the earlier documents cited by the scientists themselves in their papers.

The Science Citation Index (SCI) is the most comprehensive citation data base available today (1) and our first experiments directed at the structure of the scientific literature were performed using a quarterly compilation of the SCI (2). The overall strategy was to focus on frequently cited documents as indicators or markers of significant developments in science; to define a measure of association between pairs of highly cited documents (which we called co-citation) (3); to perform a clustering routine on all co-citeddocuments; and finally to create a spatial display of the clustered documents by multidimensional scaling or, alternatively, by drawing the cluster as a connected graph (4). In addition, an attempt was made to systematically vary the clustering level to view science either at the specialty or the disciplinary scale; and to model changes in these levels over time using numerical techniques such as INDSCAL(5).

The results of these initial experiments were striking confirmation of the importance of the internal self-ordering of scientific communications. The primary structural unit to emerge from the clustering was the narrowly defined scientific specialty, not the broader discipline or the isolated individual. Furthermore, the small clusters which came to be identified with specialties were themselves linked together in a loose network which eventually connected the entire fabric of science.

Later experiments have increased the scale of this effort by utilizing annual cumulations of the SCI. Specifically we have derived clusters from four successive annual cumulations (1970 - 1973) in order to examine the patterns of change in these networks. We will report here on a clustering of the 1973 SCI employing a slightly modified coefficient of association (normalized co-citation) which has led to considerable improvement in the number of clusters obtained. The purpose is not to set out once and for all a classification scheme for science, but rather to suggest a methodology for developing a truly dynamic classification system.

METHODOLOGY

A considerable body of evidence has been assembled in support of the contention that scientific works which become frequently cited are markers for important scientific advances. These include, for example, the original reports of discoveries, methodological advances, theoretical breakthroughs, critical experiments, and the like (6).

Since we selected these highly cited works as the basic units in our analysis, we needed a measure of association between them. The number of times they were cited together (co-citation) provided a natural and convenient measure. If S_{r_i} is the set of source items citing document r_i, and S_{r_j} the set of source items citing document r_j, then the frequency of co-citation C_{ij}, is: $C_{ij} = n(S_{r_i} \cap S_{r_j})$. This raw co-citation count was used in our earlier experiments on clustering with considerable success, provided we eliminated the co-citations involving the very high frequency documents which were invariably methodological papers.

In an attempt to avoid this somewhat artificial step, we experimented with a number of "normalizations" of the co-citation measure, all of which performed about equally well. In general, the result was that more clusters were obtained without the largest cluster exceeding a certain maximum size. The form of normalization finally selected (primarily because of its ease of interpretation) was the percentage overlap or Jaccard coefficient as it is known to numerical taxonomists (7):

$$(JC)_{ij} = \frac{n\ (S_{r_i} \cap S_{r_j}\ )}{n\ (S_{r_i} \cup S_{r_j}\ )} .$$

Because the number of highly cited documents in an annual SCI is quite large (about 16,000 documents cited 15 times or more), it was necessary to adopt a simple approach to clustering. The single-link algorithm was selected even though the effects of chaining could lead to the linking of separate subject matters rather easily. This procedure requires that a document have only one link of sufficient strength with a document already clustered for membership in the cluster. If R is a set of cited documents (a cluster) then for each $r_i \in R$ there exists at least one other document $r_j \in R$ for which $(JC)_{ij} \geq t$, where t is a pre-set threshold for the similarity coefficient.

Application of single-link clustering to the set of documents cited more than T times provides a partition of these documents into N disjoint sets or clusters. We can now define a cluster or set of source (citing) documents S_R for each cluster R such that $S_R = \bigcup S_{r_i}$ where $r_i \in R$. Citing documents may therefore be members of more than one cluster since they may cite documents in more than one cluster. Hence, clusters of citing documents are overlapping, while clusters of cited documents are non-overlapping.

DESCRIPTION OF EXPERIMENT

The clustering experiment was carried out on the magnetic tape version of the 1973 SCI. A citation frequency threshold of 15 was used and there were a total of 15,923 distinct items in the file cited 15 times or more, representing a .58% sample of all cited items. There were 431,082 citations to these distinct items, representing an 8.7% sample of all citations in the 1973 file. When this file of selected citations was inverted (i.e. placed in order by citing item rather than by cited item), a total of 140,992 distinct citing (source) items was found, which is 34.6% of all source items processed by ISI for the 1973 SCI. Since about 20% of all source items contain no citations, 43% of the source items containing citations cited at least one of the highly cited documents. This places an upper limit on the number of source documents we can hope to classify by clustering documents cited 15 times or more. The mean number of highly cited items per source item was 3, compared to a mean of 12 cited items per source for the overall file.

The next step in the processing involved permuting all highly cited items which were contained in the same source item, and then counting the number of identical pairs of cited items to determine co-citation frequency. A total of 1,244,224 pairs of cited items were created of which 713,948 were distinct pairs. Thus, the mean frequency of co-citation is 1.74. 72.1% of the pairs have a frequency of one; one pair had a frequency of 351. If every cited item had paired with every other, there would have been a total of $\frac{1}{2}n(n-1) = 126,763,003$ pairs. Since only 713,948 distinct pairs were found, the total network of cited documents is .56% connected, that is, only .56% of the possible links between documents actually existed.

Only one document out of the 15,923 items cited 15 times or more did not pair with any of the other items. Furthermore, only 9 of the documents were paired with only one other document. The mean number of links per cited document was 89.7.

In the single-link method the basic limitation is the size of the largest cluster which grows rapidly in size as the threshold is lowered. Eventually, we anticipate that all 15,923 documents minus the one unpaired document would form one gigantic cluster at the lowest level.

Single-link clustering was carried out at seven different levels of the Jaccard coefficient. The results of these runs are summarized in Table 1.

Table 1
Results of Clustering at Different Levels

level(%)	number of documents	number of clusters	mean documents per cluster	documents in largest cluster
10.0	11,537	1,003	11.5	6,293(55%)
11.6	10,464	1,246	8.4	3,657(35%)
12.6	9,896	1,367	7.2	2,021(20%)
15.0	8,537	1,554	5.5	332(4%)
16.0	8,042	1,610	5.0	117(1%)
20.0	6,208	1,609	3.9	42(.7%)
22.5	5,183	1,511	3.4	34(.6%)

Between levels 12.6% and 15% there is a sudden decrease in the size of the largest cluster from about 2,000 documents to 332. The number of clusters increases in a fairly regular manner with increase in the level, but at a point between 20% and 22.5%, the number of clusters begins to decrease with increasing level.

CLUSTERS AT THE 16% LEVEL

We will now examine in greater detail the clusters obtained at the 16% level and give some examples. The distribution of clusters by number of cited documents is given in Table 2. (Note: Since we are clustering pairs of documents, the smallest cluster obtainable contains two documents.)

Table 2
Distribution of Clusters by Number of Cited Documents

(1) number of documents	(2) number of clusters	(1) number of documents	(2) number of clusters
2	820	11	13
3	254	12	11
4	138	13	8
5	83	14	10
6	60	15	7
7	46	16	9
8	34	17	9
9	22	18	11
10	19	19	2

Table 2
Distribution of Clusters by Number of Cited Documents

(1) number of documents	(2) number of clusters	(1) number of documents	(2) number of clusters
20	1	28	1
21	2	29	3
22	3	30	2
23	1	31	2
24	1	32	2
25	2	33	4
26	5	34	1
27	3	39	1
		43 and above	20

The 1,610 clusters were then used to retrieve all the source (citing) documents from the 1973 SCI. A total of 109,147 source documents were retrieved by the clusters for a mean number of citing documents per cluster of 67.8. Since any one citing paper can appear in more than one cluster, the true number of documents classified is about 100,000, that is, roughly one-quarter of the source documents in the 1973 SCI. Presumably, this percentage classified would increase if a lower initial citation frequency threshold were used, for example, 10 times cited for the year rather than 15.

The 20 largest clusters in terms of number of cited documents are listed in Table 3.

Table 3
Twenty Largest Clusters at Level 16%

cluster number	description	number of cited items	number of citing items
2	cancer virus	117	844
56	immunology	93	982
236	muscle histochemistry	80	644
264	DNA repair	79	666
172	chromosome banding	76	757
86	C-13 NMR (biochemistry)	68	615
296	nucleoproteins	66	511
165	polypeptide hormones	62	474
259	adenylate cyclase	62	548
208	messenger RNA	52	438
36	cytochalasin-B	51	456
48	parton model	51	519
5	cyclic AMP	50	471
53	protein transport	50	407
66	lymphocytes (leukemia)	50	591
92	Australia antigen	48	413
164	vitamin-D metabolism	47	276
533	bacterial ribosomes	44	252
46	weak interactions	43	338
107	carcinogenic hydrocarbons	43	361

Names for the clusters (and for all 16% level clusters having 50 or more citing documents) were derived through an automatic title word analysis of the citing items in each cluster. In this analysis all pairs of title words were generated as well as a distribution of word displacement, that is, how many words intervene between the paired words. This information is of considerable help in determining the subject matter of each cluster, but human judgment is needed to select among the various possibilities.

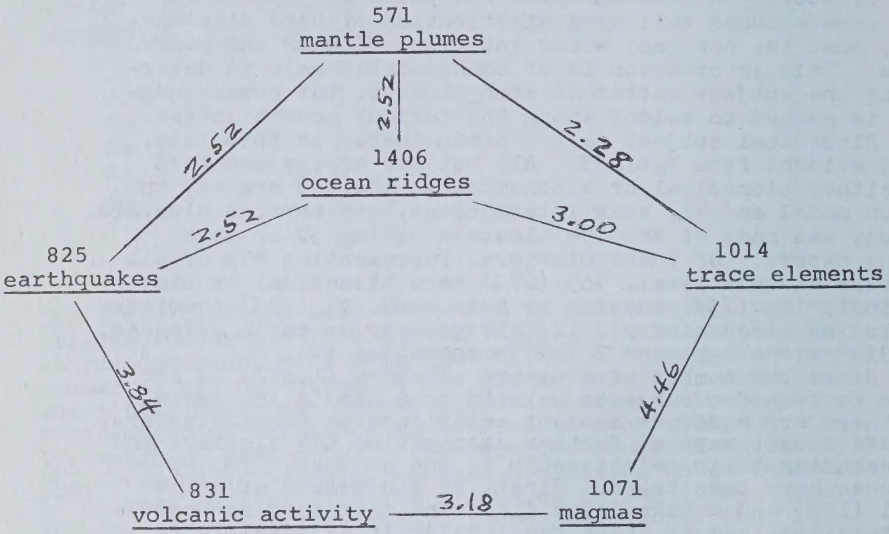
Biomedical subject matter predominates in this file, as is evident from Table 3. All but two of the clusters are either biomedical or biochemical; the two are 48, the parton model and 46, weak interactions,both physics clusters. A study was made of the 694 clusters having 50 or more citing papers. Of these clusters, representing 43% of all clusters at this level, 463 (67%) were biomedical or biochemical, 103 (15%) physics or astronomy, 91 (13%) chemistry (excluding biochemistry), 12 (2%) geology or earth sciences, 18 (3%) psychology, and 7 (1%) mathematics (8).

Since the number of clusters obtained (1,610) is too large to scan for subjects related to a user's interests, and there are numerous content relationships among clusters, we have sought ways of further aggregating the clusters and of measuring their relationship to one another. Two approaches have been taken. First, by clustering at a low level (10%) and a high level (16%) and then merging together clusters obtained at these two levels, it is possible to obtain a hierarchical ordering of the higher level clusters in the lower level clusters. For example, all the clusters on various aspects of plate tectonics formed at level 16% have coalesced into a single plate tectonics cluster at level 10%.

The second approach is to define a coupling measure between clusters. This measure has been called "cluster co-citation" (4). It is, simply, the total number of co-citations between documents in two different clusters. This measure can also be "normalized" by dividing by the number of individual linkages to obtain a mean co-citation frequency per inter-cluster link, and used as input to a clustering routine, that is, a clustering of the clusters. An example of such a "cluster of clusters" is given in Figure 1, derived at a mean cluster co-citation level of 2.25, and consisting of the major plate tectonics clusters obtained at level 16%. The cluster co-citation linkages are especially interesting because they provide us with the "cross reference" structure of the clustered file.

As an example of a cluster of cited documents and its corresponding cluster of citing (source) documents, we have selected cluster 1014, trace elements, shown in the previous plate tectonics group (Figure 1). The cluster is shown in Figure 2 as it emerges from the computer as a list of pairs of cited documents, and as a string of four documents. Figure 3 is a list (alphabetical by author) of the 27 papers which cited at least two of the four documents in 1973. Another 39 papers cited only one of the four documents for a total of 66 citing papers for the cluster.

Figure 1
Cluster of Plate Tectonics Clusters


571
mantle plumes
2.52
2.52
2.28
1406
ocean ridges
2.52
3.00
825
earthquakes
1014
trace elements
3.84
4.46
831
volcanic activity
3.18
1071
magmas

Level 2.25

DISCUSSION

We have taken a detailed look at clusters formed from a single annual SCI file. We do not want to leave the impression that we are presenting a classification scheme for science which will remain valid for all time. From studies on four successive annual files (1970 - 1973) we have found quite significant change both in the internal structure of specialties and the relationship among specialties. The magnitude and meaning of these shifts are still under study, but one conclusion cannot be avoided. It will be impossible to set up a static classification scheme for science which will continue to have validity for the science which is actually being done. What our work is leading to is a "real time" monitoring of the structural changes in science. For information retrieval this means a constantly "moving" classification scheme, with updating perhaps on a monthly or even weekly basis. We believe that this is a practical goal, attainable in the near future with available technology.

A second finding, which follows from the results presented in this paper, and which has important ramifications for information retrieval is the pre-eminence of the specialty as a structural unit in science. In our design of information retrieval systems we must shift our focus from the discipline (and perhaps from the individual user) to the closely knit specialty group. This should not be interpreted as encouraging divisiveness among researchers since, as we have seen, each specialty is located in a web of inter-specialty linkages. But in our concentration on the needs of very broad groups or at the other extreme on idiosynchratic individuals, we have perhaps missed an opportunity to design systems in an optimum way. We believe that the most successful system for ordering scientific literature will be the one which captures most precisely the internal social and intellectual structure of science; or more accurately, the one which reflects the social and intellectual context within which scientific documents are generated. This context, with its complex web of associations, is, we believe, most accurately reflected in the patterns of citation in scientific literature.

ACKNOWLEDGMENTS

The work of the first author was supported by Grant GS-37211 from the National Science Foundation, and the work of the second author by Grant 5-R01 LM00911-02 from the National Library of Medicine. We also wish to thank Louis Holmes of the Institute for Scientific Information for systems and programming.

REFERENCES

1. E. Garfield,"'World Brain' or 'Memex?' Mechanical and intellectual requirements for universal bibliographic control," in: Foundations of Access to Knowledge (Syracuse: Syracuse University Press, 1968) 169-196.
2. H.G. Small and B.C. Griffith, "The structure of scientific literatures I: Identifying and graphing specialties," Science Studies 4: 17-40 (1974).
3. H.G. Small, "Co-citation in the scientific literature: a new measure of the relationship between two documents," Journal of the American Society for Information Science 24 (4): 265-269 (1973).
4. B.C. Griffith, H.G. Small, J.A. Stonehill and S. Dey, "The structure of scientific literatures II: Toward a macro- and microstructure for science," Science Studies 4 (October, 1974) to be published.
5. J.D. Carroll and J.J. Chang, "Analysis of individual differences in multidimensional scaling via an N-way generalization of 'Eckart-Young' decomposition," Psychometrika 35 (3): 283-319 (1970).
6. H.Small, "Characteristics of frequently cited papers in chemistry," final report on NSF contract C-795.
7. P.H.A. Sneath and R.R. Sokal, Numerical Taxonomy: The Principles and Practice of Numerical Classification (San Francisco: W. H. Freeman, 1973).
8. A list of these clusters is available from the authors.

Figure 2

Printout and Graph for Cluster 1014
"Trace Elements"

```
07/12/74                                      CLUSTER   1.014                                                          PAGE   852

                   CITATION 1                    FREQ 1                        CITATION 2                      FREQ 2   STRENGTH

GAST PW         GEOCHIM COSMOCHIM AC  32105768     34     KAY R            J GEOPHYS RES           75158570      34      283
GAST PW         GEOCHIM COSMOCHIM AC  32105768     34     SCHNETZLER CC    GEOCHIM COSMOCHIM AC  34 33170        16      190

KAY R           J GEOPHYS RES         75158570     34     GAST PW          GEOCHIM COSMOCHIM AC  32105768        34      283

PHILPOTTS JA    GEOCHIM COSMOCHIM AC  34 30770     16     SCHNETZLER CC    GEOCHIM COSMOCHIM AC  34 33170        16      185

SCHNETZLER CC   GEOCHIM COSMOCHIM AC  34 33170     16     GAST PW          GEOCHIM COSMOCHIM AC  32105768        34      190
SCHNETZLER CC   GEOCHIM COSMOCHIM AC  34 33170     16     PHILPOTTS JA     GEOCHIM COSMOCHIM AC  34 30770        16      185

                                 TOTAL RECORDS IN CLUSTER     =          6

                                 NUMBER OF DISTINCT PAIRS     =          3

                                 NUMBER OF REFERENCES         =          4

                                 PERCENTAGE CONNECTEDNESS     =      50.00

                                 AVERAGE YEAR UNIQUE CITATION=     1969.50

                                 AVERAGE FREQ UNIQUE CITATION=       25.00

                                 AVERAGE STRENGTH OF CLUSTER =      219.33
```

```
             .283                .190                        .185
Kay R  -----------  Gast PW  ----------  Schnetzler CC  ----------  Philpotts JA
```

Figure 3
Co-citing Papers for Cluster 1014

```
CLUSTER   SOURCE AUTHOR    SOURCE JOURNAL   VOL   PAGE   CODE   FREQ       ***   T I T L E   ***

1014      BANNO S          CHEM GEOL         11      1            2
                FORMULATION OF PARTITION-COEFFICIENTS FOR TRACE-ELEMENTS DISTRIBUTION BET*EEN MINERALS AND MAGMA
1014      BES*ICK AE       GEOCH COS A       37    183            2
                EXPERIMENTAL STUDY OF ALKALI-METAL DISTRIBUTIONS IN FELDSPARS AND MICAS
1014      CAMPSIE J        NATURE-PHYS      244     71            2
                CHEMISTRY OF THOLEIITES FROM REYKJANES RIDGE AND CHARLIE GIBBS FRACTURE ZONE
1014      CAPALDI G        EARTH PLAN        17    247            2
                RARE-EARTH ABUNDANCES IN ALKALINE VOLCANIC-ROCKS FROM CAMPANIA. SOUTH-ITALY
1014      CHURCH SE        GEOL S AM B       84    *31            2
                LEAD AND STRONTIUM ISOTOPIC STUDIES IN CASCADE MOUNTAINS - BEARING ON ANDESITE GENESIS
1014      CULLERS RL       GEOCH COS A       37   1499            3
                EXPERIMENTAL STUDIES OF DISTRIBUTION OF RARE-EARTHS AS TRACE-ELEMENTS AMONG SILICATE MINERALS AND LI
                QUIDS AND *ATER
1014      DRURY SA         CHEM GEOL         11    167            2
                GEOCHEMISTRY OF PRECAMBRIAN GRANULITE FACIES ROCKS FROM LE*ISIAN COMPLEX OF TIREE. INNER HEBRIDES. S
                COTLAND
1014      FLO*ER MFJ       CHEM GEOL         12     81            2
                TRACE-ELEMENT DISTRIBUTION IN LAVAS FROM ANJOUAN AND GRANDE-COMORE. *ESTERN INDIAN-OCEAN
1014      GALE GH          EARTH PLAN        18     22            2
                PALEOZOIC BASALTIC KOMATIITE AND OCEAN-FLOOR TYPE BASALTS FROM NORTHEASTERN NE*FOUNDLAND
1014      HART SR          EARTH PLAN        20    201            3
                SUBMARINE BASALTS FROM KILAUEA RIFT HA*AII - NONDEPENDENCE OF TRACE-ELEMENT COMPOSITION ON EXTRUSION
                 DEPTH
1014      HELMKE PA        GEOCH COS A       37   1513            3
                RARE-EARTH ELEMENTS. CO. SC AND HF IN STEENS MOUNTAIN BASALTS
1014      JENSEN BB        GEOCH COS A       37   2227            2
                PATTERNS OF TRACE-ELEMENT PARTITIONING
1014      MCCARTHY TS      EARTH PLAN        18    433            3
                ORIGIN OF EUCRITES AND DIOGENITES
1014      MONTIGNY R       GEOCH COS A       37   2135            2
                TRACE-ELEMENT GEOCHEMISTRY AND GENESIS OF PINDOS OPHIOLITE SUITE
1014      MUELLER PA       GEOL S AM B       84   3645            2
                SECULAR CHEMICAL VARIATION IN A SERIES OF PRECAMBRIAN MAFIC ROCKS. BEARTOOTH-MOUNTAINS. MONTANA AND
                *YOMING
1014      NAGASA*A H       CONTR MIN P       39    301            2
                RARE-EARTH DISTRIBUTION IN ALKALI ROCKS FROM OKI-DOGO-ISLAND. JAPAN
1014      ONIONS RK        EARTH PLAN        19    397            3
                PETROGENETIC RELATIONSHIPS OF ACID AND BASIC ROCKS IN ICELAND - SR-ISOTOPES AND RARE-EARTH ELEMENTS
                IN LATE AND POSTGLACIAL VOLCANICS
1014      ONIONS RK        NATURE           243    213            2
                STRONTIUM ISOTOPES AND RARE-EARTH ELEMENTS IN BASALTS FROM HEIMAEY AND SURTSEY VOLCANIC ERUPTIONS
1014      SCHEIDEG.KF      EARTH PLAN        17    397            2
                VOLCANIC ASH LAYERS IN DEEP-SEA SEDIMENTS AND THEIR PETROLOGICAL SIGNIFICANCE
1014      SCHEIDEG.KF      J GEOPH RES       78   3340            2
                TEMPERATURES AND COMPOSITIONS OF MAGMAS ASCENDING ALONG MID-OCEAN RIDGES
1014      SCHILLIN.JG      NATURE           242    565            2
                ICELAND MANTLE PLUME - GEOCHEMICAL STUDY OF REYKJANES RIDGE
1014      SCHILLIN.JG      NATURE-PHYS      242      2            2
                AFAR MANTLE PLUME - RARE-EARTH EVIDENCE
1014      SHA* DM          CAN J EARTH        9   1577            3
                DEVELOPMENT OF EARLY CONTINENTAL CRUST .1. USE OF TRACE-ELEMENT DISTRIBUTION COEFFICIENT MODELS FOR
                PROTOARCHEAN CRUST

1014      SINHA AK         GEOCH COS A       37   1823            2
                ISOTOPIC EVOLUTION OF COMMON LEAD
1014      THOMPSON G       CHEM GEOL         12     99            2
                TRACE-ELEMENT DISTRIBUTIONS IN FRACTIONATED OCEANIC ROCKS. .2. GABBROS AND RELATED ROCKS
1014      *ILDEMAN TR      GEOCH COS A       37    419            3
                RARE-EARTHS IN PRECAMBRIAN SEDIMENTS
1014      *YLLIE PJ        TECTONOPHYS       17    189            2
                EXPERIMENTAL PETROLOGY AND GLOBAL TECTONICS - A PREVIE*
* TOTAL CLUSTER TITLES    66
```

THE EFFECT OF TERMINAL SPEED ON SEARCH PATTERNS IN INTERACTIVE INFORMATION RETRIEVAL

ROGER K. SUMMIT
Lockheed Missiles & Space Company, Inc.
Palo Alto, California 94304

ABSTRACT

Little attention has been given to differences in on-line search patterns employed by individuals using high-speed terminals (480 characters per second) versus low-speed terminals (30 characters per second). Questions regarding the cost effectiveness of the more expensive high-speed, leased-line terminals and lower-cost, low-speed dial-up terminals are studied, using the DIALOG™ system, with a finding that the cost of high-speed searching, at a level of about 21 to 26 hours per month of searching, is about equal to that of low-speed searching.

INTRODUCTION AND BACKGROUND

Lockheed Information Systems has operated a commercial, computer-based, on-line, information retrieval service using a language called DIALOG™ since 1969.(1) In the early days of DIALOG™ system operation, most customers used a high-speed terminal consisting of a cathode-ray-tube (CRT) display that operated at a rate of 240 characters per second (cps) via a special leased-line network, and an auxiliary printer operating at 30 cps. In 1971, we introduced a direct dial and, later, a dial-up capability, via the TYMSHARE network, both of which support teletype-compatible terminals (usually hard copy) that normally operate at 30 cps. In 1973, the speed of the high-speed network was increased from 240 to 480 cps. The service currently supports a large number of terminals; some of these terminals operate at 30 cps via the dial-up network, and others operate at 480 cps via the leased-line network.

It has been our observation that the approach to answering a search question when using an interactive computer terminal varies with respect to the speed of the terminal used. For example, a user who has learned to search on a high-speed terminal tends to maintain and use the search habits thus developed when utilizing a low-speed terminal, and vice versa.

Another significant consideration that often arises relates to the comparative cost effectiveness of using a high-speed CRT terminal versus a low-speed hard-copy terminal. This paper presents a preliminary analysis of the difference in searching cost for the two types of terminals, assuming a common search pattern is involved. To our knowledge, this question has not been previously addressed from the standpoint of an operational system environment.

AVERAGE SEARCH

A search or search question is defined as a collection of commands issued by the searcher to the system. Differences in searches can be characterized by differences in the frequency of usage of various commands during the course of the search. For purposes of cost evaluation, a model consisting of a standard search pattern will be developed, based on actual command-use averages by a sample of ERIC* users. The statistics to be utilized have been drawn from an as-yet unpublished report by the Institute of Library Research, University of California.(2)

---

*ERIC is an acronym identifying a data base produced by the National Institute of Education and used by many educational institutions.

Critical command classes in terms of usage and cost variation consist of the following:

- File Output Commands — These commands are EXPAND, which displays a portion of the index of searchable words and DISPLAY (or TYPE), which outputs the citation and abstract to the users terminal.

- Set Commands — A set is created by a SELECT or COMBINE command and corresponds to each word or phrase, or Boolean combination thereof requested in the search.

The variation in usage of other classes of commands for the various types of terminal is insignificant. In one sense, the file output commands are a measure of the amount of interaction between the users and the file, and the number of sets is a measure of the completeness or complexity of treatment of the search.

A total of 772 search questions (542 from high-speed terminals and 230 from low-speed terminals) containing over 20,000 commands from all types of terminals were analyzed. The average or standard search question contained the following mix of commands:

	File Output Commands	Set or Search Commands
Average Number of Commands	10.2	11.6

The average search time for all questions was 16.7 minutes, of which 9.56 minutes were used for the output commands. Considering the average time per command class by user type (i.e., high-speed versus low-speed terminal), and extending that observation by the number of commands per question in the average search, we can derive the average difference in search time for high-speed versus low-speed terminals as shown in Table 1.

Table 1. Average Difference in Search Time for High-Speed Versus Low-Speed Terminals

High Speed Terminals	File Output Commands	Set Commands	Total
Average time/command (sec)	36.1	26.0	—
Average time/question (sec)	368.2	301.6	669.8
Low Speed Terminals			
Average time/command (sec)	64.9	30.6	—
Average time/question (sec)	662.0	355.0	1017.0
Difference			
Average time/command (sec)	28.8	4.6	—
Average time/question (sec)	293.8	53.4	347.2

The total difference per standard search question can be seen to be 347.2 seconds or 5.79 minutes. (It should be noted that the times used in computing these results measure the time from the request of a command to the input of the next command and thus include "think" as well as "keyboard" time.) If we assume that there is no general difference among users with regard to the command mix entered, then the difference between low-speed terminal times and high-speed terminal times should represent an approximation of the actual response-time differences between the two terminals.

COST ANALYSIS

In addition to search charges of $25 per hour for the ERIC file, users of high-speed terminals pay a fixed lease cost of approximately $600 per month for both equipment rental and communications, regardless of the extent of usage, whereas dial-up terminals users pay a surcharge of $10 per hour for use of the TYMSHARE data communications network, plus a terminal cost of approximately $100. The difference in fixed monthly cost of the two types of terminals is thus approximately $500 per month.

Considering only variable search costs (i.e., excluding terminal rental and the monthly charges for high-speed terminals), there is an hourly search charge of $25, and a TYMSHARE surcharge of $10 for dial-up terminals which leads to the following costs per search question, using the figures previously presented:

	Search Charge (@ $25 per hour)	TYMSHARE (@ $10 per hour)	Total
High-Speed Terminals			
Cost per Question	$4.651	–	$4.651
Low-Speed Terminals			
Cost per Question	$7.062	$2.825	$9.887
Difference			
Cost per Question	$2.411	$2.825	$5.236

At a cost difference of $5.24 per question, these results would suggest that anyone who searches more than 95 questions a month, involving about 26 hours of searching per month, clearly would find it more economical to use a high-speed terminal. In this analysis, consideration has not been given to the labor cost savings associated with the lower search times on the high-speed terminal, which even under a modest assumption of $15 per hour (including overhead), increases the above cost difference to $6.69 per question and the breakeven point to 75 questions per month or about 21 hours of searching per month.

CONCLUSIONS

Although the habits of searchers are affected by the speed of the terminal they use, a high-speed terminal — on the basis of an average search, as derived, and the cost differences indicated for using high-speed versus low-speed terminals — appears to show a cost advantage at between 21 and 26 hours per month of searching. Additional studies should be conducted relative to the qualitative differences in the search results with regard to terminal type and the relative advantages for different types of files.

REFERENCES

1. R. Donati, "Information Center Profile," Information-Part 1, Vol. 5, No. 2, March — April 1973, pp. 114—117
2. Institute of Library Research, "Analysis of ERIC On-Line File Searching Procedures and Guidelines for Searching," Univ. of Calif., Berkeley, Calif., September 1974

LOGIC NETWORK SYNTHESIS USING DIGITAL-SUMMATION THRESHOLD-LOGIC GATES.

S.L.HURST,
University of Bath, BATH, BA2 7AY, England.

ABSTRACT

The recently-developed Digital-Summation Threshold-Logic ("DSTL") gate will be introduced, and its advantages and disadvantages briefly discussed in comparison with previous analogue-type threshold-logic gates. Some typical applications will be shown indicating package and pin savings in random logic networks. Consideration of optimum universal package specifications for random logic work will be mentioned.

1. INTRODUCTION

Threshold-logic gates are memoryless, multiple-binary-input, single- or multiple-binary-output circuits that operate by an arithmetic summation process to determine the 0 or 1 gate output state(s). The relationship between the binary inputs per gate and a resultant binary output z may be expressed as:

$$\text{output } z = 1 \text{ if } \sum_{i=1}^{n} a_i x_i \geqslant t,$$

$$= 0 \text{ if } \sum_{i=1}^{n} a_i x_i < t,$$

where x_i, $i = 1$ to n, are the n gate inputs, each of which may take the value 0 or 1,

a_i, $i = 1$ to n, are "weights" associated with each respective gate input x_i,

t = gate output threshold value, $0 \leqslant t \leqslant \sum_{i=1}^{n} a_i$.

It is usually taken without loss of generality that the input weights a_i and the gate threshold value t are all positive integer values. Figure 1 illustrates the usual mechanical analogy of the action of a single-output threshold-logic gate.

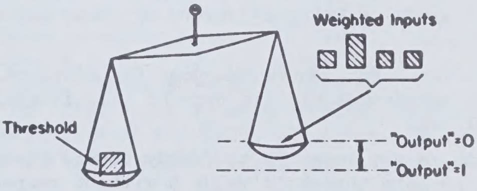


Fig.1. Mechanical analogy of a threshold-logic gate.

Conventional Boolean AND and OR are but extreme cases of such threshold relationships. The AND relationship is:

$$\text{gate output } z = 1 \text{ if } \sum_{i=1}^{n} a_i x_i = \sum_{i=1}^{n} a_i,$$

$$= 0 \text{ otherwise,}$$

whilst the OR relationship is:

$$\text{gate output } z = 1 \text{ if } \sum_{i=1}^{n} a_i x_i \geqslant 1,$$

$$= 0 \text{ otherwise.}$$

NAND and NOR relationships merely interchange the gate output 0 or 1 value in each definition.

The class of Boolean functions known as *linearly-separable* are also termed threshold-logic functions, as each such function may be realized by the use of a single threshold-logic gate, of appropriate input weights and threshold value. The mathematical study of linear separability in all its aspects constitutes a major part of the research work in the threshold-logic field during the past two decades[1,2,3,4], such work largely outweighing practical developments. However, it has been clear from early days that if threshold-logic gates were economically available, then useful savings in the number of gates and interconnections per system could be proposed. Examples of such design savings will be given later in this paper.

Present-day digital system design may embrace the following three general areas; firstly the area where complex tailor-made integrated-circuit chips can be justified - the pocket calculator market for example; secondly the area where m.s.i. and l.s.i. packages may be extensively used - the digital computer, telecommunications and instrumentation area for example, and thirdly the simple but widely encountered random-logic area, where s.s.i. logic packages are currently employed. This third area may not be as sophisticated as the first two, and in general does not occupy the attention of learned societies and powerful research teams. Nevertheless it is the proposition of this paper that this is the area where threshold logic should be widely considered, rather than in computer or similar fields which has been the general case to date. This is not to decry in any sense the vast amount of research done in considering the application of threshold logic to computer applications - reduction in circuit response time may be a very significant feature in this area, but this may be considered as the other end of the logic application spectrum in comparison with the area under consideration here.

2. THE CIRCUITS OF THRESHOLD-LOGIC GATES

Previous threshold-logic circuits involved some form of analogue-summation technique within the gate structure. Details will be found in published literature[5,6,7,8]. The monolithic circuits were generally characterised by very fast gate propagation times but ECL-type logic levels, these characteristics being a feature of their non-saturating emitter-coupled circuitry.

However for general-purpose (as distinct from specific) applications, the disadvantages of this type of circuit realization in monolithic form are as follows:

(a) it is not possible to freely change the gate input weightings and/or the gate threshold value t without respecifying the ohmic value of internal resistors;

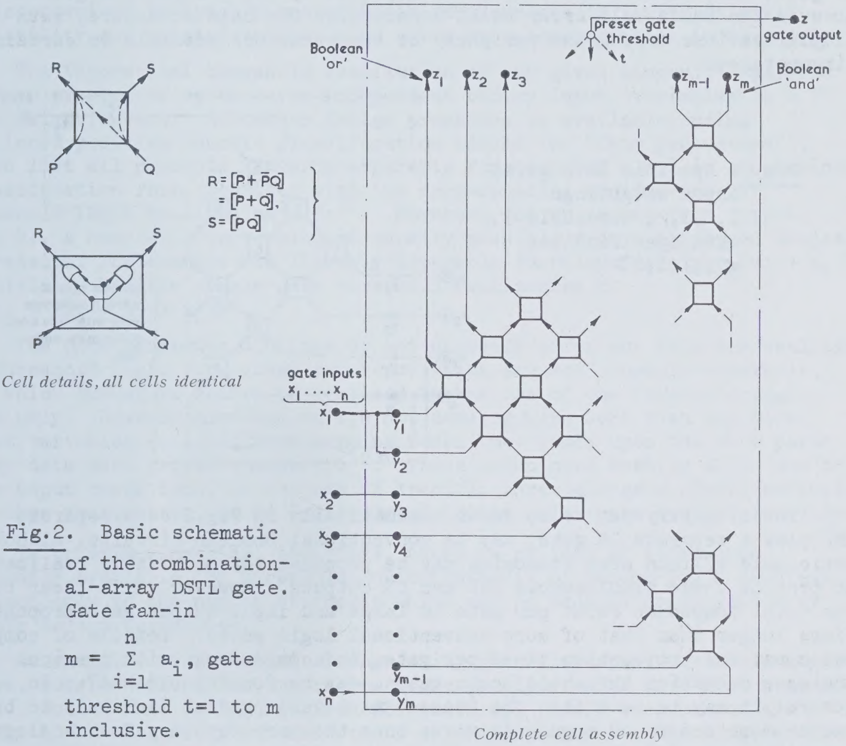
(b) it is not possible to simultaneously provide more than one threshold value t per gate,

and

(c) the complete gate circuit is subject to component tolerances which eventually limit the fan-in and threshold discrimination that can economically be realized.

As an alternative approach to the previous analogue-summation threshold-logic gates, an entirely digital construction recently has been proposed[9].

This construction eliminates disadvantages (a) to (c) above, but has itself one major disadvantage, that of relatively slow overall gate propagation times. However for the general-purpose industrial market, it is contended that high speed is not a vital parameter. This digital-construction gate has been termed a "digital-summation threshold-logic" (or "DSTL") gate. Its action and advantages will be briefly recounted.



Cell details, all cells identical

Complete cell assembly

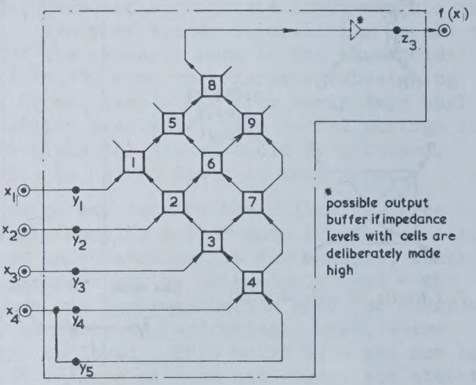
Fig.2. Basic schematic of the combinational-array DSTL gate. Gate fan-in $m = \sum_{i=1}^{n} a_i$, gate threshold t=1 to m inclusive.

Figure 2 illustrates the basic arrangement of a DSTL gate. It consists of a regular array of identical cells, its action being that a single logic 1 signal on any line y_i, i = 1 to m, cascades diagonally through the array to appear at the upper output line z_1. With two (or more) logic 1 inputs on lines y_i, then two (or more) output lines z_i will be 1-energised. The output lines z_i regularly "fill up" from z_1 to z_m as correspondingly more logic 1 inputs are present, irrespective of the point of application of these 1 inputs, and regularly "empty" from right to left as 1 inputs are removed.

In the general case illustrated in Fig.2, the output line z_1 corresponds to the Boolean OR of the applied inputs, whilst output z_m corresponds to the Boolean AND. In total, the z_i output lines realize all the threshold detection values t from 1 to m.

The cell input lines y_1 to y_m each have a natural input weighting of 1. Therefore to provide a gate input weighting of, say, 2, any two such lines may be metalized together, for example as shown in Fig.2 for gate input x_1.

For specific gate specifications, truncation of this complete triangular array is possible. Figure 3 illustrates a case where a threshold value t=3 only is required, with input weightings of 1,1,1,2. Further internal simplifications are also possible in many such cases; for example cell 4 in Fig.3 is in fact redundant, cells 8 and 9 could be replaced by one 3-input OR gate, and cells 1 and 5 could be replaced by one 3-input AND gate. However the basic cell array still constitutes the main structure, with simplifications around the periphery of the structure possible in certain circumstances.



* possible output buffer if impedance levels with cells are deliberately made high

Fig.3. Specific DSTL gate, input weightings 1,1,1,2, threshold 3, gate specification $< 1,1,1,2 >_3$

The circuitry per cell, shown schematically in Fig.2 as a separate AND plus a separate OR gate, may be conventional Boolean circuits, although noticeable silicon area economies may be proposed for monolithic realization to provide these simultaneous AND and OR outputs. However it is clear that the total component count per gate is large and input-output gate propagation times longer than that of more conventional logic gates. Details of component count and propagation times per gate, and comparisons with previous analogue-summation threshold-logic gates, may be found published[9]. In general it may be said that the (speed x power) product is likely to be one or more orders of magnitude worse than the previous very fast analogue-summation gates[7,8].

The total component count per gate is not a serious constraint in monolithic form. On the other hand, the regular cell array structure is potentially advantageous, as it raises the possibility of being able to fabricate one basic chip design with a one or more groups of cells, final metalization then being used to select the array(s) mecessary to produce the required DSTL package specification(s). It is along these lines that further research work is currently being pursued, with a view to finally producing a commercially-viable general-purpose DSTL package. For immediate research work, a monolithic 24-pin DSTL package is current being made, this evaluation package containing one large array, specification $< 2,2,1,1,1,1 >_t$, t = 1 to 8 inclusive, all eight outputs z_1 to z_8 and their complements $\overline{z}_1$ to $\overline{z}_8$ being individually available.

3. LOGIC DESIGN USING DSTL GATES

Pursuing the proposition that it is the area of random logic which may constitute the widest field for threshold logic, other areas being served by conventional MSI, LSI or custom-built monolithic circuits *, a review of random-logic requirements is pertinent.

By the very definition of random logic, any complexity of logic function may be encountered. However when industrial problems are surveyed, it is seldom found that more than five or six binary inputs are present per net or logic area. An average of 3·2 binary inputs per random logic problem has been quoted. Grouping of a digital system into this size tends to occur during the formulation of the design, possibly because it may be difficult or inconvenient to specify random-logic requirements in larger "blocks". Testability and maintainability also enter into consideration in this area.

The theoretical threshold realization of any given linearly-separable Boolean problem of up to seven independent binary input variables ($n \leqslant 7$) is straightforward. A look-up design procedure is available, using published positive canonic classification tables (or "Chow parameters"), which list all possible linearly-separable functions of $n \leqslant 7$ in a numerical classification form, together with the corresponding minimum-integer threshold-logic realization[1,10,11]. However, for up to say four inputs, ($n \leqslant 4$), a hueristic conversion is usually possible from a minimised Boolean expression; for example the linearly-separable function $f(x)=[x_1\bar{x}_2(x_3 + \bar{x}_4)]$ intuitively has the single-gate threshold realization of $\langle 2x_1 + 2\bar{x}_2 + x_3 + \bar{x}_4 \rangle_5$.

The Chow parameter listings do not directly offer any help for realizing in threshold-logic form given functions which are *not* linearly-separable, and which therefore cannot be realized by the use of one threshold-logic gate only. However provided we are not dealing with more than say five input variables ($n \leqslant 5$), then mapping techniques based upon the Chow parameter data have proved successful[12]. These techniques readily allow don't-care input conditions, or the use of specific threshold-gate specifications, or other constraints, to be incorporated in the multi-gate design procedure.

To briefly summarise, therefore, there is no lack of techniques currently available for threshold-logic synthesis of random-logic problems, subject only to the number of inputs which may be handled conveniently by simple design techniques.

As examples of potential savings using general-purpose threshold-logic gates, consider the following random logic problems which have been encountered. Notice that design requirements which are widely encountered are likely to be adequately catered for by existing packages - decoding, arithmetic operations, parity checking/generation, and the like can call upon appropriate MSI or LSI packages. Thus the examples quoted below are problems possibly unique to a particular job. Overall, the parameter of increasing significance to the system designer is becoming total I/C package *pin count*, this parameter correlating well with total equipment cost, where total cost is defined as the sum of design, component, assembly, testing, and maintenance costs.

Example 1.

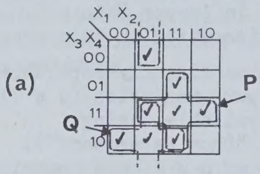
The following non-linearly-separable 4-input, one-output problem was given:

$$f(x) = [\, x_1(x_2x_4 + x_3x_4) + \bar{x}_1\bar{x}_4(x_2 + x_3) + x_2x_3 \,]$$

---

* There is of course no reason why analogue-summation or digital-summation threshold-logic subsystems should not be incorporated to form some inner part of a "conventional" Boolean MSI or LSI package. The user of such a package could remain in ignorance of this feature without disadvantage, in the same way that he may not know the exact logic structure of present-day packages.

By using mapping techniques[12], the realization shown in Fig.4 was developed. A conventional Boolean realization of this problem would employ about 9 vertex gates, including the two input inverters, with a total pin count of around 35. The pin count in Fig.4(c) is 15.



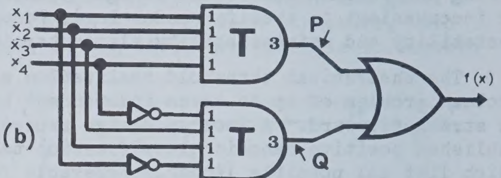
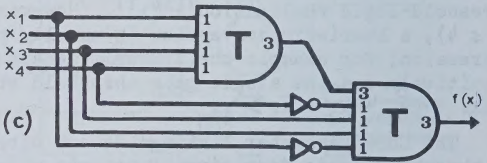


Fig.4. Example No.1, function $f(x) = [\,x_1(x_2x_4 + x_3x_4)$
$+ \bar{x}_1\bar{x}_4(x_2 + x_3) + x_2x_3\,]$
a) Karnaugh map of function
b) first threshold-logic realization
c) subsequent simplification.



However in this problem it is possible to further simplify Fig.4(c), as it may be noted that the threshold partitions P and Q are similar except for the inverted inputs of x_1 and x_4. Figure 5 illustrates a better final solution, this having a pin count of 9. Notice also that using such techniques input inverters are often eliminated.

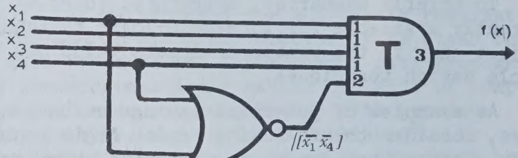


Fig.5. Final simplification of Fig. 4.

Example 2.

The following problem was given:

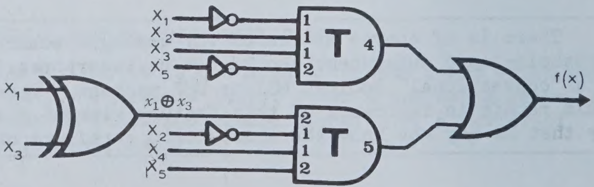
$$f(x) = [\,x_1\bar{x}_2\bar{x}_3x_5 + x_1\bar{x}_3x_4x_5 + \bar{x}_1x_2\bar{x}_5 + \bar{x}_1x_3\bar{x}_5 + x_2x_3\bar{x}_5 + \bar{x}_1\bar{x}_2x_3x_5 + \bar{x}_1x_3x_4x_5\,]$$

By mapping or factorizing, this may be re-expressed in a more compact form:

$$f(x) = [\,\bar{x}_5\{\bar{x}_1x_2 + \bar{x}_1x_3 + x_2x_3\} + x_5\{x_1\bar{x}_3(\bar{x}_2 + x_4) + \bar{x}_1x_3(\bar{x}_2 + x_4)\}\,],$$

from which the realization shown in Fig.6 develops. The final OR gate of this circuit may be incorporated in one or other of the two threshold gates if there are sufficient gate inputs available, but even as shown the pin count has been reduced to 22 from the figure of about 36 which would be present in corresponding vertex gate realization.

Fig. 6. Circuit realization of Example 2.



Example 3.

A certain arithmetic processor required two outputs, given by:

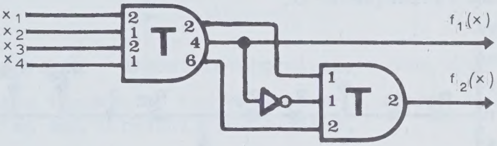
$$f_1(x) = [\, x_1(x_3 + x_2x_4) + x_2x_3x_4 \,],$$

and

$$f_2(x) = [x_1(\bar{x}_2\bar{x}_3 + \bar{x}_3\bar{x}_4 + x_2x_3x_4) + \bar{x}_1(\bar{x}_2x_3 + x_3\bar{x}_4 + x_2\bar{x}_3x_4) \,] .$$

The use of threshold gates with multiple threshold outputs, such as that provided by the DSTL gate, proves advantageous for this problem. One solution obtained by threshold mapping is shown in Fig.7. Note that no complements of the inputs are necessary in this solution. The total pin count is 13 compared with about 37 for a comparable Exclusive OR and vertex gate realization.

Fig.7. Circuit realization for Example 3.

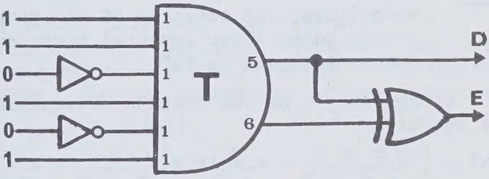


Example 4.

The problem given was to detect one particular 6-bit word, allowing one bit to be in error. With a one-bit error present, and indication of this fact was desired.

Again using the multi-threshold facility, a simple solution can be proposed. Figure 8 shows the solution, assuming the 6-bit word to be detected was 110101. The total pin count in this solution is 15, compared with about 30 for a solution using Exclusive OR and vertex gates.

Fig.8. Circuit for 6-bit word detector, word = 110101. Output D = word detected, output E = one bit error present.



4. FURTHER RANDOM-LOGIC DESIGN TECHNIQUES

The examples above show how conventional vertex gates and Exclusive-OR gates may profitably be employed around a threshold-logic gate. In these examples, the final realization evolved largely by intuitive ideas around the threshold logic.

Exclusive-OR relationships, however, are extremely powerful. To some extent, linearly-separable functions and exclusive-OR functions may be regarded as "opposite" each other, as the exclusive-OR relationship is the most inconvenient relationship to realize using threshold-logic (or vertex) gates. Mapping of exclusive-OR functions will readily illustrate why this is so. However a formal design technique which is capable of handling both exclusive-OR *and* threshold-logic relationships would be a very powerful design tool.

A method which currently is being researched in this area is the use of Rademacher-Walsh spectral techniques[1,13,14]. Here is not the place to detail this subject; suffice to say that network synthesis is undertaken in the spectral domain rather than the usual Boolean domain, a given logic problem first being converted by means of a fast Walsh transform[15] into a series of integer numbers (the "spectrum") which uniquely define the given function as would a Boolean expression or Boolean truth-table. Spectral translation operations may now be carried out on the spectrum, to re-arrange the spectral coefficient values if necessary into some preferred ordering, each translation in this spectral domain rigourously corresponding to an interchange of input variables or the introduction of exclusive-OR operators in the Boolean domain. Preliminary details of these techniques will be found published[16,17]. The feature of these techniques is to tend to realize traditionally complex networks with a simple vertex or threshold-logic "core", prefaced and/or succeeded by one or more exclusive-OR gates. A further feature of these spectral methods is that they are amenable to automatic or interactive CAD, and are not constrained by the limitations of mapping techniques.

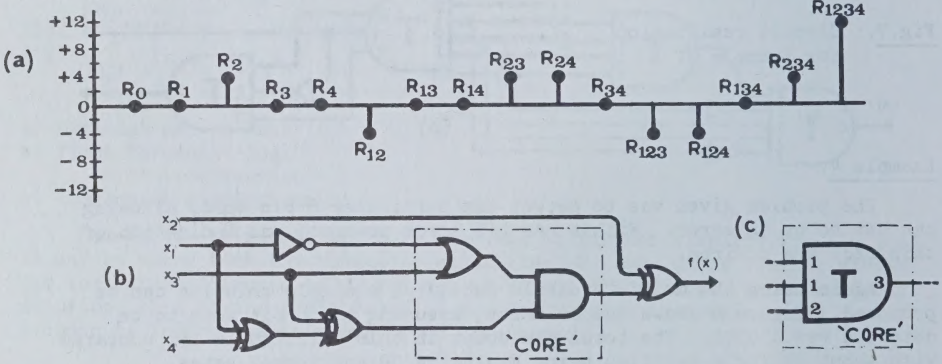


Fig.9. Example network synthesis obtained by spectral translation techniques; (a) spectrum of the given function, (b) "core" of vertex gates after spectral translation, (c) equivalent threshold-logic "core".

As an example of the results obtainable, Fig.9 gives a realization for the function

$$f(x) = [\, \bar{x}_1\bar{x}_2(\bar{x}_3x_4 + x_3\bar{x}_4) + x_3x_4(\bar{x}_1x_2 + x_1\bar{x}_2) + x_1x_2(\bar{x}_3 + \bar{x}_4) + x_2\bar{x}_3\bar{x}_4 \,] \ .$$

Note that the inclusion of exclusive-OR gates eliminates the need for the inversion of certain inputs to be provided.

These newer ideas now coming into evidence are further pointers to the potential of non-vertex gates in logic synthesis to supplement the present-day almost universal use of the logically-inefficient NAND or other vertex gates.

5. UNIVERSAL THRESHOLD-LOGIC PACKAGES

The question of what constitutes an optimum threshold-logic package for general-purpose usage has been considered[18,19]. The canonic Chow parameter classification tables come into consideration in this area. However, it currently seems probable that more than one general-purpose package, possibly three, may be justified. With the regular array structure of the DSTL gate, one master chip layout may be possible, with different metalization to give the particular package specifications.

Existing proposals have tended to concentrate on 14- or 16-pin d.i.l. packages. Fig.10 illustrates one 16-pin package which has been proposed. The presence of two (or more) gates per package is desirable in order to generally minimise unused package connections.

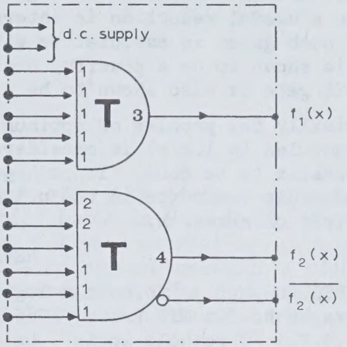


Fig.10. Suggested 16-pin d.i.l. general-purpose threshold-logic package, containing one majority gate $\langle 1,1,1,1,1 \rangle_3$, and one $n \leqslant 4$ "universal" gate $\langle 2,2,1,1,1,1 \rangle_4$ with true and complementary outputs.

However existing proposals require urgent re-appraisal in the light of:

a) the increasing emphasis upon the use of the exclusive-OR operator in conjunction with both vertex and threshold-logic gates,

and

b) the recent "stretch" of the standard 16-pin plastic d.i.l. package to a 20-pin version with the same package width and pin spacings.

For example a package similar to that illustrated in Fig.10 but incorporating say two exclusive-OR gates in addition to the threshold gates may prove widely applicable. Full statistical research on random logic networks is necessary in this area to clarify this problem, and (ideally) suggest an optimum universal logic module.

6. UNIVERSAL LOGIC MODULES.

The search for a commercially-viable universal logic module ("u.l.m.") or package which is capable of realizing every possible combinational logic function of n inputs, where n is some relatively small number, has occupied the attention of several authorities. Current contenders in the field are:

a) certain cellular-array logic structures[20,21],
b) multiplexer circuits[22,23],
and
c) random-access-memory circuits and their variations[24].

To this list should now be added the threshold-logic package, augmented by vertex and/or exclusive-OR gates.

In all such possibilities the utilisation of active devices within the u.l.m. is small for any specific application, although ideally the package pin utilisation should be high. Following on from the previous section, research to compare the viability of a threshold-logic u.l.m. in comparison with the three main alternatives has yet to be undertaken. Some preliminary random checks, however, indicate that a threshold-logic-based u.l.m. may involve a less complex chip than the alternative propositions, with resultant higher manufacturing yield possibilities.

7. CONCLUSIONS

This paper has introduced the digital-summation threshold-logic (DSTL) gate, a new development in the field of threshold logic. This development gives emphasis to the possible use of threshold logic, particularly when (i) gate propagation times are not critical, and (ii) available MSI or LSI packages are not appropriate.

The use of threshold-logic gates in random logic applications is known to give a useful reduction in interconnections (pin count) per network. The use of such gates in association with vertex and (particularly) exclusive-OR gates is shown to be a powerful design tool. The multi-output capability of the DSTL gate is also shown to be a powerful advantage in certain circumstances.

Finally the problem of optimum package specifications and universal logic modules (u.l.m's) is considered. Considerable useful research in this area remains to be done. It is hoped that the disclosure of the DSTL gate as a possible candidate in the u.l.m. field may add impetus and interest to this field of research.

REFERENCES

1. DERTOUZOS,M.L.: "Threshold logic: a synthesis approach", Research Monographs No.32, MIT Press, 1965.
2. HU,S.T.: "Threshold logic", University of California Press, 1965.
3. WINDER, R.O.: "Fundamentals of threshold logic", Air Force Cambridge Scientific Report AFCLR-68-0066, 1968.
4. MUROGA,S.: "Threshold logic and its application", Wiley Interscience,1971
5. AMODII,J.J.: et.al.: "An integrated threshold gate", Digest I.S.S.C.C., N.Y., 1967, pp.114-115.
6. COHEN,S. & WINDER,R.O.: "Threshold gate building blocks", Trans.IEEE, C.18, 1969, pp.816-823.
7. BEINART,J., et.al.: "Integrated threshold logic", Air Force Avionics Laboratory Report AFAL-TR-69-195, 1969.
8. HAMPEL,D., et.al.: "Threshold-logic implementation of a modular computer system design", NASA Report CR-1668, 1970.
9. HURST,S.L. "Digital-summation threshold-logic gates: a new circuit element", Proc.I.E.E., 120, 1973, pp.1301-1307.
10. WINDER,R.O.: "Threshold functions through n = 7", AFCLR Report No.7,1964.
11. HURST,S.L.: "Threshold logic: an engineering survey", Mills & Boon, London, 1971.
12. HURST,S.L.: "Synthesis of threshold-logic networks using Karnaugh-mapping techniques", Proc.IEE, 119, 1972, pp.1119-1128.
13. WALSH,J.L.: "A closed set of normal orthogonal functions", Amer. Journal Maths., 45, 1923, pp.5-24.
14. COLEMAN,R.P.: "Orthogonal functions for the logical design of switching circuits", Trans.IRE., EC.10, 1961, pp.379-383.
15. SHANKS,J.L.: "Computation of the fast Fourier transform", Trans.IEEE, EC.18, 1969, pp.457-459.
16. EDWARDS,C.R.: "The application of the Rademacher-Walsh transform to digital circuit synthesis", Proc.Symp. Theory & Application of Walsh Functions, Hatfield, England, 1973.
17. EDWARDS,C.R.: "The application of the Rademacher-Walsh transform to Boolean function classification and threshold-logic synthesis", Trans.IEEE, C.23, (to be published shortly).
18. COHEN,S. & WINDER,R.O.: "Threshold-gate building blocks", Trans.IEEE, C.18, 1969, pp.816-823.
19. HURST,S.L.: "Specification of threshold-logic gates for optimum s.s.i. logic packaging", Electronic Letters, 8, 1972, pp.514-515.
20. AKERS,S.B.: "A rectangular logic array", Trans.IEEE, C.21, 1972, pp.848-857.
21. MUZIO,J.C. & MILLER,D.M.: "A powerful cellular array", Proc. Third Manitoba Conf. on Numerical Maths., 1973, pp.315-322.
22. ANDERSON,J.L.: "Multiplexers double as logic circuits", Electronics, 42, 1969, pp.100-105.
23. YAU,S.S. & TANG,C.K.: "Universal logic modules and their application", Trans. IEEE, C.19. 1970, pp.141-149.
24. KRAMME,F.: "Standard read only memories simply complex logic design", Electronic, 43, 1970, pp. 88-95.

STATISTICAL ANALYSIS OF THRESHOLD LOGIC CIRCUITS

C. R. BAUGH AND B. A. WOOLEY
Bell Laboratories
Holmdel, New Jersey 07733

ABSTRACT

A first-order statistical analysis of a differential current-switching circuit implementation of threshold logic is described. It is shown that a threshold of $T = 8.5$ *and* $\sum w_i x_i \leq 17$ *can be realized with an ECL compatible power supply of -5.2V and the tolerances achievable with conventional integrated circuit diffusion processing. The analysis supported the design of a monolithic 4-out-of-9 detection circuit which was fabricated.*

INTRODUCTION

The economics of interconnections motivate an increasing scale of circuit integration in digital systems. Consequently, much of the logic for such systems will be transferred from the interchip environment of small scale integration to a quieter and more tightly controlled intrachip environment. Threshold logic circuits offer one means of exploiting this transfer.

In this paper a general differential current-switching circuit configuration is proposed for the implementation of threshold logic, and the results of a first-order statistical analysis of the configuration's static behavior are presented. The proposed circuit configuration is intended to both permit realization of the maximum threshold value achievable via a current-switching approach and also maintain compatibility with conventional emitter-coupled logic circuits. The differential circuit configuration was used to fabricate a monolithic M-out-of-N detection circuit[1]. The statistical analysis relates the maximum threshold to the circuit power supply voltages and the tolerances of the integrated circuit components.

CURRENT-SWITCHING THRESHOLD LOGIC

The following definition is assumed for a threshold logic function. A Boolean function $f(\vec{x})$ of N binary variables, $\vec{x} = (x_1,\ldots,x_N)$ is a threshold function if there exists a vector of integer weights $\vec{w} = (w_1,\ldots,w_N)$ and a threshold $T \triangleq T' - 0.5$, where T' is an integer, such that

$$f = \begin{cases} 1 & \text{if } \vec{w}\cdot\vec{x} = \sum_{i=1}^{N} w_i x_i > T \\ 0 & \text{if } \vec{w}\cdot\vec{x} < T \end{cases} \tag{1}$$

The basic operations of threshold logic are the formation of the weighted sum $\vec{w}\cdot\vec{x}$ and the comparison of that sum to the threshold T. Current switching is an effective means of performing these operations within an integrated circuit.[2,3] Shown in general form in Fig. 1 is the differential current-switching circuit configuration proposed for the implementation of threshold logic. In this circuit the threshold logic operations are relatively insensitive to common-mode circuit variations due to the differential nature of the configuration.

In the circuit of Fig. 1 N weighted input current switches are connected to the pair of sum nodes S_A and S_B. The output currents of these switches are summed in the nominally identical resistors R_{1A} and R_{1B} and establish the voltages at the sum nodes. These voltages are then level shifted and used to drive one or more theshold current switches differentially. The value T is established either by adjusting the relative magnitudes of the level-shifting voltages, V_{LA} and V_{LB}, or by introducing a sum node offset current. Threshold current switches with differing threshold values can be driven from a single pair of sum nodes either by connecting additional level-shifting networks to the sum nodes or by using a level-shifting network with multiple outputs.

The level shifts, V_{LA} and V_{LB}, in Fig. 1 are implemented as shown in Fig. 2 for V_{LA}. Transistor Q_{L4A} is included in the level-shifting network so as to match the IR voltage drops in these networks as closely as possible to the voltages across the summing resistors R_{1A} and R_{1B}. The current sources in Figs. 1 and 2 are conventional transistor sources with resistive emitter degeneration as shown in Fig. 3.

The voltages at the sum nodes S_A and S_B in Fig. 1 are given approximately by

$$V_{SA} \simeq V_{CC} - \left[\sum_{i=1}^{N} \bar{x}_i w_i\right] I_u R_{1A} \tag{2}$$

and

$$V_{SB} \simeq V_{CC} - \left[\sum_{i=1}^{N} x_i w_i\right] I_u R_{1B}, \tag{3}$$

where I_u is the nominal unity weight source current. The differential voltage applied to the threshold switch is

$$\begin{aligned} V_o &= (V_{SA}-V_{LA}) - (V_{SB}-V_{LB}) \\ &\simeq 2\left[\sum_{i=1}^{N} x_i w_i - \frac{1}{2}\left(\sum_{i=1}^{N} w_i + n_{2A} - n_{2B}\right)\right] I_u R_1, \end{aligned} \tag{4}$$

where n_{2A} and n_{2B} are integers and $R_1 = R_{1A} = R_{1B}$. The threshold value T corresponds to the hypothetical condition where $V_o = 0$, i.e.

$$T = \frac{1}{2}\left(\sum_{i=1}^{N} w_i + n_{2A} - n_{2B}\right) \tag{5}$$

The resistors upon which the differential voltage V_O depends are all realizable as combinations of nominally identical resistors with resistance R_1. This results in the best possible component matching and, together with the differential approach, provides for minimum circuit sensitivity to processing and operating environment.

STATISTICAL ANALYSIS

The practical utility of threshold logic circuits is determined largely by the maximum threshold value that can be realized and the total number of input current switches that can be connected to sum nodes S_A and S_B. The first-order statistical analysis was performed to determine the mean, $\overline{V}_O$, and standard deviation, σ_{V_O}, of the differential voltage V_O as a function of power supply voltages and the tolerances of the integrated circuit components.

The analysis was conducted under the following assumptions.

1. All statistically varying independent parameters y are normally distributed with mean $\overline{y}$ and normalized standard deviation $\sigma_{yN} = \sigma_y/\overline{y}$.

2. All resistances except R_I are series/parallel combinations of resistors with nominal resistance R_1.

3. All transistors are assumed to be nominally identical. They are characterized by their current-gain, β, and forward biased base-emitter voltage to collector current transfer function, $I_C \simeq I_s \exp[qV_{BE}/kT]$.

4. The absolute value of the input voltage V_i is the same for all of the input current switches and is denoted by V_I. Variations in V_i are not considered and as a result $\overline{V}_O$ and σ_{V_O} are given as a function of V_I.

5. Second and higher order terms of quantities much less than 1 are neglected.

6. Changes in the mean unity weight current, $\overline{I}_u$, are accomplished by changing R_I while maintaining the remainder of the component values constant.

The results of the analysis show that $\overline{V}_O$ and σ_{V_O} are insensitive to $\sigma_{\beta N}$. Large variation in transistor current gains will not be a factor in the circuit performance. $\overline{V}_O$ and σ_{V_O} are also insensitive to the number of threshold switches driven from one sum node pair. For practical circuits the number of threshold switches would not be a limiting factor in the design.

From assumption 4, variations in V_i are not included in the expressions from $\overline{V}_O$ and σ_{V_O}. However, the results show that V_I must be greater than .1V to reliably switch the current in the input current switches so as to prevent serious degradation of $\overline{V}_O$.

As expected σ_{RN} and σ_{I_sN} are the dominant factors in σ_{V_O}. Fig. 4 shows σ_{V_O} as a function of σ_{RN} and σ_{I_sN} for the typical component values shown in Table I. The components are described in terms of their mean and standard deviation. The power supply voltages are the typical ECL levels of V_{CC} = 0.0V and V_{EE} = -5.2V. From the curves of Fig. 4, $\sigma_{V_o} \leq 32$ mV if $\sigma_{RN} \leq 1\%$ and $\sigma_{I_sN} \leq 10\%$. These values of σ_{RN} and σ_{I_sN} are typical of results obtainable with standard diffused processing.[4,5]

To establish the maximum threshold value two design criteria must be considered. One is the minimum $\overline{V}_o$ required to ensure reliable switching of the threshold current switch while maintaining high circuit yield. This is reflected as $\overline{V}_o = \gamma\sigma_{V_o}$ + 100 mV. The second criterion is whether the threshold current switch is used either as an input current switch for a subsequent sum node pair or as part of a conventional ECL output stage. This condition determines the maximum voltage swing at the sum node that can be partitioned into increments of I_uR_1. The allowable swing is largest when the threshold detector is used in an ECL output stage. Table II tabulates the maximum realizable threshold values for the two design criteria and various parameter values. From Table II, the maximum threshold is 6.5 for the threshold switch as an input current switch and 8.5 for the threshold switch part of conventional ECL output stage when typical component values and component value variations (σ_{RN}=.01, σ_{I_sN}=.1) are assumed.

The analysis also demonstrates that at least 17 input current switches can be connected to a sum node without significantly effecting $\overline{V}_o$ or σ_{V_o}. If the sum node is clamped to prevent saturation of the input current sources, a threshold logic element with $\Sigma w_i x_i \leq 17$ and a $T \leq 8.5$ can be realized with standard integrated circuit processing. Such a threshold logic element is a powerful logic element.

CONCLUSION

A first-order statistical analysis of a differential current-switching circuit realization of threshold logic was performed. The results demonstrate that σ_{RN} and σ_{I_sN} are the dominant factors. The results also demonstrate that threshold logic elements with $\Sigma w_i x_i \leq 17$ and $T \leq 8.5$ can be realized with present-day, standard diffusion integrated circuit processing. The analysis supported the design of a monolithic, integrated 4-out-of-9 circuit that was fabricated.

REFERENCES

1. B. A. Wooley and C. R. Baugh, "An Integrated M-Out-of-N Circuit Using Threshold Logic," IEEE Journal of Solid-State Circuits," Vol. SC-9, October 1974.

2. J. J. Amodii, R. O. Winder, D. Hampel and T. R. Mayhew, "An Integrated Threshold Logic Gate," ISSCC Digest Tech. Papers, pp. 114-115, 1967.

3. D. Hampel and R. O. Winder, "Threshold Logic," IEEE Spectrum, Vol. 8, pp. 38-39, May 1971.

4. H. H. Stellrecht and G. Kelson, "A Self-Contained 10-Bit Monolithic D/A Converter," ISCC Digest of Technical Papers," Vol. 16, pp. 16-17, February 1973.

5. G. Kelson, H. H. Stillbrecht and D. S. Perloff, "A Monolithic Digital-to-Analog Converter Using Ion Implantation," IEEE J. Solid-State Circuits, Vol. SC-8, pp. 396-403, December 1973.

$\overline{R}_1 = \overline{R}_E = 500\Omega$

$\overline{I}_s = 10^{-15}A$

$R_I = 8k\Omega$

$\overline{\beta} = 100$

$\sigma_{\beta N} = 20\%$

$\overline{V}_{CC}-\overline{V}_{EE} = 5.2V$

$\sigma_{(V_{CC}-V_{EE})} = 10mV$

$\sum_{i=1}^{N} w_i = 17$

TABLE I Parameters for analysis of Fig. 4

$\sigma_R/\overline{R}$	$\sigma_{I_s}/\overline{I}_s$	σ_{V_o} (mV)	T_{max} (γ=4, ECL Out)	T_{max} (γ=4, Sum Node Out)
.005	.05	17	12.5	9.5
.005	.1	21	11.5	8.5
.01	.1	32	8.5	6.5
.015	.1	46	6.5	5.5
.02	.2	64	5.5	4.5

TABLE II Maximum threshold

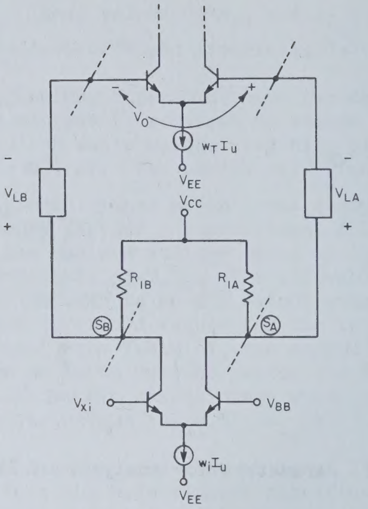

-
+
VO
wT Iu
VEE
-
VLB
+
-
VLA
+
VCC
RIB
RIA
SB
SA
Vxi
VBB
wi Iu
VEE

FIG. 1 Differential current-switching threshold logic circuit

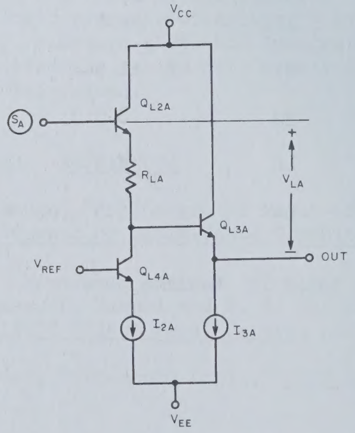

VCC
SA
QL2A
RLA
+
VLA
QL3A
OUT
VREF
QL4A
I2A
I3A
VEE

FIG. 2 Level shift network for V_{LA}

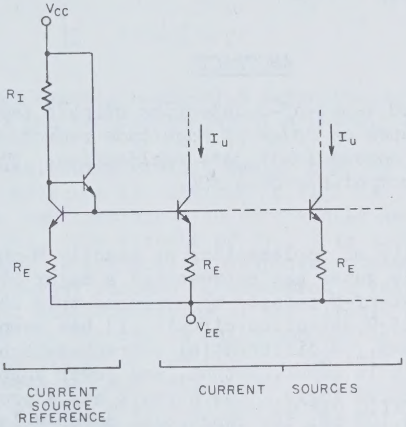

VCC
RI
Iu
Iu
RE
RE
RE
VEE
CURRENT SOURCE REFERENCE
CURRENT SOURCES

FIG. 3 Current sources

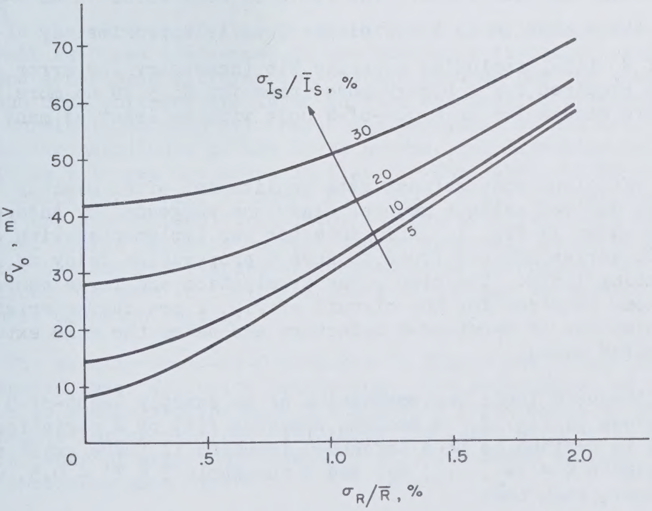


FIG. 4 σ_{V_o} as a function of σ_{RN} and σ_{I_SN}

M-OUT-OF-N DETECTION USING THRESHOLD LOGIC

B. A. WOOLEY AND C. R. BAUGH
Bell Laboratories
Holmdel, New Jersey 07733

ABSTRACT

An integrated 4-out-of-9 detection circuit implemented using threshold logic achieves an order of magnitude reduction in power-delay product relative to a conventional gate realization. The circuit is signal and power supply compatible with ECL.

The difficulty of implementing an exactly M-out-of-N detector with conventional logic gates has represented a major obstacle to the more widespread use of M-out-of-N codes. To overcome this obstacle, an integrated exactly 4-out-of-9 detection circuit [1] has been designed using a threshold logic approach. A differential current-switching configuration is used and the circuit is input, output, and power supply compatible with commercial emitter-coupled logic. The circuit has a propagation delay of 16 nsec and dissipates 100 mW; its functional power-delay product of 1600 pJ is substantially below that achievable with a conventional gate design.

M-out-of-N codes are characterized by the requirement that each code word contain M ones and N-M zeros. These codes have a number of advantages including (1) any single bit error is detectable, (2) all multiple bit unidirectional errors are detectable, (3) decoding is accomplished simply with an M-input AND gate, (4) timing for asynchronous operation can be extracted from the code words. The ratio of code words in an M-out-of-N code to the number in an N-bit binary code is approximately $\sqrt{\frac{2}{\pi N}}$. Therefore, if N' bits, including a parity bit (necessary for error detection), are required for a binary code, then for $N' \leq 39$ no more than $N' + 2$ bits are needed for an M-out-of-N code with at least as many code words.

An efficient conventional gate realization of an exactly 4-out-of-9 detector, derived using a general algorithm suggested by Anderson and Metze [2], is shown in Fig. 1. This detector was implemented with commercial 10,000 series ECL and found to have a propagation delay of 12 nsec while dissipating 1.5 W. The high power dissipation and large number of interconnections required for the circuit of Fig. 1 are characteristic of gate implementations of M-out-of-N detectors and deter the more extensive use of M-out-of-N codes.

A threshold logic implementation of an exactly 4-out-of-9 detector is shown in Fig. 2. A Boolean function $f(\vec{x})$ of N variables $\vec{x} = (x_1, \ldots, x_N)$ is defined to be a threshold function if there exist a vector of integer weights $\vec{w} = (w_1, \ldots, w_N)$ and a threshold $T \triangleq T' - 0.5$, where T' is an integer, such that

$$f(\vec{x}) = \begin{cases} 0 & \text{if } \vec{w}\cdot\vec{x} = \sum_{i=1}^{N} w_i x_i < T \\ 1 & \text{if } \vec{w}\cdot\vec{x} > T \end{cases}$$

As indicated by Fig. 2, exactly M-out-of-N detection requires weights $w_i = 1$ ($i = 1, \ldots, N$) and two thresholds $T_1 = M - 0.5$ and $T_2 = M + 0.5$.

The current-switching circuit design used to realize the threshold logic 4-out-of-9 detector is outlined in Fig. 3. Current-switching, as a general approach, has been shown to be a viable means of implementing threshold logic [3], [4]. The circuit of Fig. 3 is distinguished from previously reported work in a number of respects. Threshold logic is used in an externally transparent fashion to implement a relatively complex function. The terminal characteristics of the circuit are typical of 10,000 series ECL, and the circuit is fully compatible with this family of logic. Emphasis in the design of the circuit has been placed on minimizing the sensitivity to processing and environment. It is intended that the circuit be realizable at high yields with standard diffusion processing. A differential approach is used internally to achieve the best possible circuit balance and the maximum voltage margins obtainable for a given threshold value. Also, the internal voltages critical to the summation and comparison operations of threshold logic depend primarily on nominally identical resistors, which permit the best possible resistance matching. A first-order statistical analysis indicates that the circuit approach of Fig. 3 could reasonably be used, with diffused resistors, for the detection of codes as large as 7-out-of-15, containing 6435 code words.

In the circuit of Fig. 3 the currents from nine unity-weight input current switches are summed in the nominally identical resistors R_{1A} and R_{1B}, and generate a differential voltage between the sum nodes S_A and S_B. The sum node voltages are level shifted and used to differentially drive two threshold current switches. The threshold values are established by the relative magnitudes of the level shifts; relationships between the level shifting voltages are given in Fig. 3. The level shifts are implemented as indicated in Fig. 4, where $R_1 = R_{1A} = R_{1B}$. Current sources in the circuit are conventional transistor sources with resistive emitter degeneration. Three separate current source reference networks are used to establish the desired temperature dependencies throughout the circuit. A clamp circuit is used to limit the sum node voltage excursions outside the threshold decision region.

The exactly 4-out-of-9 detector of Fig. 3 was integrated with standard buried layer diffusion processing. The performance of this realization is summarized in Table I along with performance of an experimental NOR-gate implementation. The threshold logic 4-out-of-9 detector achieves an order of magnitude reduction in power-delay product relative to the conventional gate circuit.

The threshold logic circuit approach outlined in Fig. 3 is on-chip compatible with conventional emitter-coupled logic, and represents a powerful addition to conventional current-switching circuit techniques. As applied to M-out-of-N detection, the threshold logic approach results in a dramatic reduction in power dissipation, interconnections, and device count.

REFERENCES

1. Wooley, B. A. and Baugh, C. R., "An Integrated M-Out-of-N Circuit Using Threshold Logic," IEEE Journal of Solid-State Circuits, Vol. SC-9, Oct. 1974.

2. Anderson, D. A. and Metze, G., "Design of Totally Self-Checking Check Circuits for M-Out-of-N Codes," IEEE Trans. Computers, Vol. C-22, pp. 263-269, March, 1973.

3. Amodei, J. J., Winder, R. O., Hampel, D. and Mayhew, T. R., "An Integrated Threshold Logic Gate," ISSCC Digest Tech. Papers, Vol. X, pp. 114-115, 1967.

4. Hampel, D. and Winder, R. O., "Threshold Logic," IEEE Spectrum, Vol. 8, pp. 32-39, May, 1971.

	Threshold Logic 4-out-of-9	NOR-gate 4-out-of-9
Propagation Delay	16 nsec	12 nsec
Power Dissipation	100 mW	1,500 mW
Power-Delay Product	1600 pJ	18,000 pJ

TABLE I

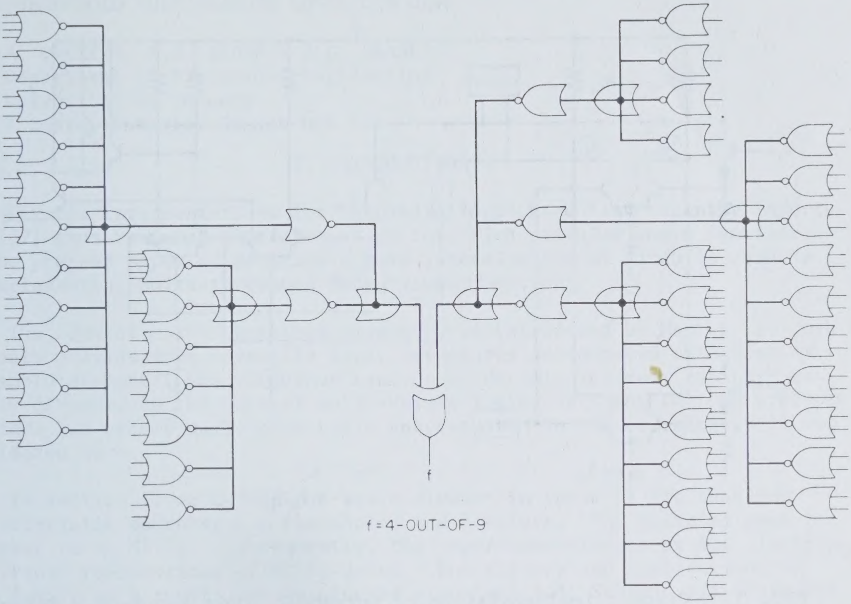


FIG. 1 Conventional gate 4-out-of-9 detector

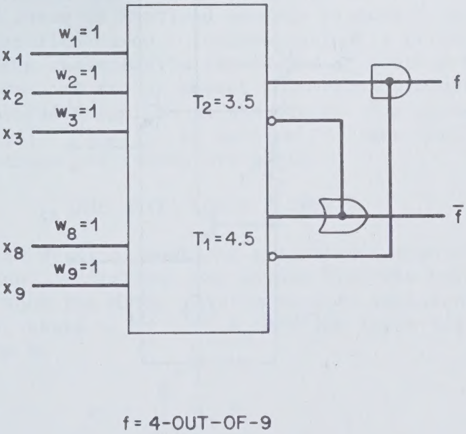

$w_1 = 1$
x_1
$w_2 = 1$
x_2
$w_3 = 1$
x_3
$T_2 = 3.5$
f
$\bar{f}$
$w_8 = 1$
x_8
$T_1 = 4.5$
$w_9 = 1$
x_9
f = 4-OUT-OF-9

FIG. 2 Threshold logic 4-out-of-9 detector

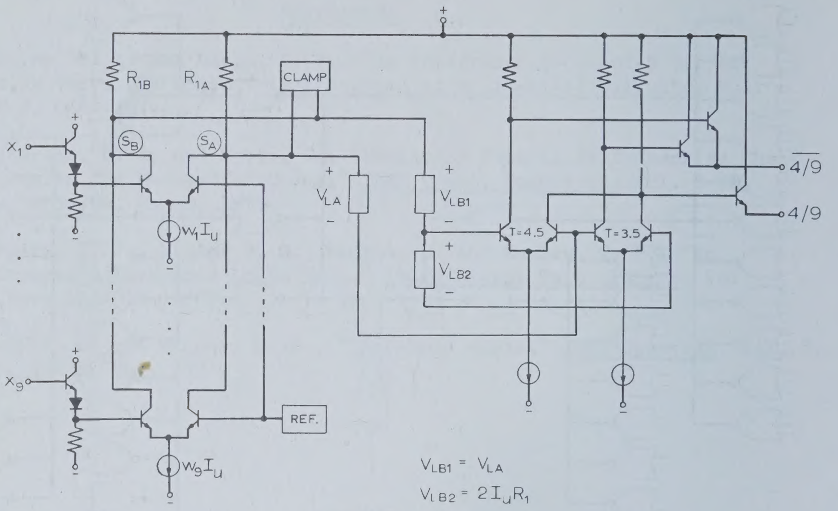

+
R_{1B}
R_{1A}
CLAMP
x_1
S_B
S_A
+
V_{LA}
−
+
V_{LB1}
−
+
V_{LB2}
−
W_1I_u
T=4.5
T=3.5
$\overline{4/9}$
4/9
x_9
REF.
W_9I_u
$V_{LB1} = V_{LA}$
$V_{LB2} = 2I_uR_1$

FIG. 3 Circuit implementation of threshold logic 4-out-of-9 detector

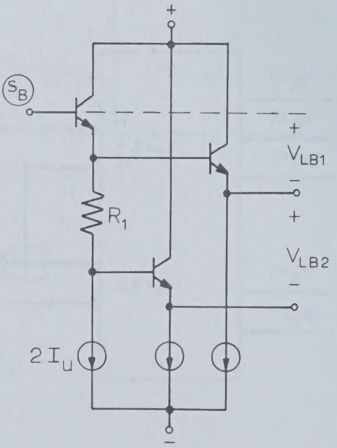

+
S_B
+
V_{LB1}
−
R_1
+
V_{LB2}
−
$2I_u$
−

FIG. 4 Typical level shifting network

MULTI-THRESHOLD MULTI-VALUED LOGIC CIRCUITS

A. DRUZETA, A.S. SEDRA & Z.G. VRANESIC
Department of Electrical Engineering
University of Toronto
Toronto, Ontario, Canada M5S 1A4

I. INTRODUCTION

Circuit implementations for threshold logic have been recently reported [1,2] thus renewing the interest in this rich class of logic families. In the present paper we describe a more general class of circuits capable of implementing multi-threshold many-valued functions.

The idea of multi-threshold elements was introduced by Haring [3].Such elements provide more versatile logic structures as compared with single-threshold elements. On the other hand, considerable interest has been recently directed to the area of multi-valued logic. Marrying the two concepts together has proved quite profitable and resulted in the circuits which are considered here.

In section II we define our basic element in terms of its transfer characteristic possessing m thresholds and R values. The basic element is referred to as MT(R). Subsequently, the paper concentrates on the electronic circuit realizations of MT(R) gates. The algebra and applications of this family of circuits are considered elsewhere [4]. Section III discusses the basic circuit implementation principle based on current-mode logic. A practical circuit is considered in detail and experimental results as well as results obtained with computer simulation are presented.

Section IV presents another current-mode circuit implementation requiring only NPN transistor, which is more suitable for integrated circuit (IC) fabrication. Perhaps the most difficult part of multi-valued logic is the realization of storage elements. An attempt in this direction is described in section V where an R-valued storage element is defined in a block form and its detailed circuit implementation is given. To add variety in our presentation the circuits considered in section V are of the voltage-mode type. We note at the outset that these storage elements have been built and successfully applied in the design of a pseudo-random sequence generator [5][6]. Finally, in section VI, some conclusions are drawn and few projections and trends are given.

II. THE MT(R) LOGIC ELEMENTS

Fundamentally, an MT(R) element has a nonlinear transfer characteristic h(e) where e, the input excitation, may assume discrete values in the set $\{0,1,2,\ldots,e_{max}\}$. When the MT(R) gate is used to implement a logic function $F(x_1,x_2,\ldots,x_n)$, where $x_1,x_2,\ldots,x_n$ are the input logic variables,the excitation e is given by,

$$e = \sum_{i=1}^{n} a_i x_i$$

with a_i denoting the weight associated with x_i. It follows that,

$$F(x_1,x_2,\ldots,x_n) = h(e)$$

Also, all the variables x_i as well as the functions F and h assume discrete values in the set {0,1,2,...,R-1}. Finally we note that if the total number of thresholds is m then, for $a_i > 0$,

$$m \le e_{max} = (R-1) \sum_{i=1}^{n} a_i$$

III. CURRENT-MODE IMPLEMENTATION OF MT(R) ELEMENTS

Having defined the MT(R) element in section II, we now consider its bipolar transistor circuit implementation. An example of a transfer characteristic h(e) and its circuit realization is shown in Fig. (1). The input variables, properly weighted, are represented by current signals that are multiples of a basic current increment I_o(1 ma for the present example). The current summation is performed at the low-impedance input node with the current sum (eI_o) providing the gate excitation.

A bias current I_B is generated by resistor R_1 together with the diode-connected transistor Q_1. This bias current is essential to prevent the saturation of the transistors connected to Q_{18} for the case e = 0. Simple reasoning also indicates that to accomplish switching at the middle points between thresholds the bias current should be, $I_B = 1.5\ I_o$. The sum of $(e\ I_o + I_B)$ is reproduced in resistor R_2 through the action of the current mirror Q_1-Q_2. The voltage thus developed at node S is compared with the threshold voltages through the differential-pair switches Q_{10}-Q_{11}, Q_{12}-Q_{13}, Q_{14}-Q_{15} and Q_{16}-Q_{17}. Note that the number of switches is equal to the number of thresholds m. The threshold voltages are generated by the voltage divider R_4 to R_8 together with Q_9. Furthermore, to obtain low source impedances,generated voltages are buffered by the emitter follower chain, Q_{20} to Q_{23}. The voltage gap V_G between two consecutive threhold voltages is chosen equal to I_oR_2. (In our example $V_G \approx 0.7$ volts).

Depending on the value of the excitation e one transistor in each differential pair will be ON with the other being OFF. The appropriate collector terminals depending on the h(e) required are grouped together and connected to the output current-mirror Q_{18}-Q_{19}. The current at the collector of Q_{19} will be the required $h(e)I_o$. Note that the basic current increment I_o is internally generated, by R_3 and Q_3, and is used to bias all the differential-pair switches through a multiple-output mirror formed by Q_3 to Q_7. If jumps in the output current level exceeding one increment are required the appropriate switch should be biased with a multiple of I_o equal to the desired number of increments at the discontinuity. These multiple currents can be simply obtained by paralleling transistors or, if IC technology is adopted, by using transistors of multiple areas. Similarly, the output variable $h(e)\ I_o$, can be assigned any required weight by adjusting the area of Q_{19} relative to that of Q_{18}.

In the example considred, the unused collectors of the differential-pair switches are connected to the power supply. If desired, these collectors may be connected to an additional output mirror providing another characteristic function.

We note that changing the grouping of the collectors connected to the output mirror changes the realized h(e), and makes the design more flexible. Note also that multiple outputs are possible by adding PNP transistors in the same manner as Q_{19}. Thus, the current-mode gate has one input node and multiple output terminals; a dual to the situation with voltage-mode circuits.

Performance analysis of the current-mode implementation

The quality of the gate transfer characteristic depends on the gain in the switching region. Since the current mirrors have a current-gain of almost unity, the gain of the gate can be determined by considering only one of the differential pairs. Simple analysis shows that for the value of the gap voltage $V_G \cong 0.7^V$ the switching gain $\simeq 6.5$. The gain can be increased by using a higher value for the threshold gap.

To investigate the effects of parameter tolerances on the performance of the gate we consider a gate realizing the stair-case unary function E(x) and its complement N(x). Both these functions are chosen since all logic levels are utilized at both the input and the outputs. Thus, errors will have a tendency to accumulate providing a worst-case situation. The function E(x) is realized by the basic circuit if the collectors of all the right-hand-side transistors of the differential pairs are connected to the output summing junction. The complement function N(x) is simultaneously obtained by connecting the collectors of the left-hand-side transistors to another output summing junction.

As far as the effects of a fractional change in V_{CC} is concerned we note that if all gates are fed by the same power supply then I_o and I_B will change by almost the same fraction as V_{CC}. However, since the threshold voltages are also power-supply dependent a compensating effect results.

If all the transistors of one gate are fabricated on the same IC chip then their parameters will track with changes in ambient temperature. In the prototype circuits constructed, IC transistor arrays were utilized and the circuits were partitioned among the different packages in a manner that ensures minimum temperature dependence of h(e). The temperature tracking of IC resistors also causes I_o and I_B in the different parts of the system to proportionally track. The most drastic temperature effects though are caused by the changes in the V_{BE} drops of the emitter followers. These changes manifest themselves as changes in the threshold voltages in the gate static characteristic. Experimental results are shown in Fig. (2).

The successful operation of the gate circuit in a logic network depends on the equality of the basic current I_o among different gates. Clearly LSI technology would be extremely beneficial in this situation since all the biasing resistors will be on the same IC chip and thus well matched. Simple analysis of the gate implementing E(x) shows that the correct operation of a 5-valued system requires a resistor matching of better than ±5%.

Another source of error is the nonunity current-gain of the multiple-output current-mirror. Emitter degeneration resistors help in reducing the effects of the V_{BE} mismatches while simultaneously increasing the mirror output resistance. Also, improved mirror designs reduce the effects of the finite transistor β. This will be illustrated below.

Variations on the current-mode circuit

The basic gate shown in Fig. (1) cannot realize an arbitrary h(e). If h(o) is smaller than the number of negative transistions in the transfer characteristic then a stacking of differential pairs is required. The circuit of Fig. (3) illustrates the concept of stacking. Note that for $e < i$ the V_{BE} voltage drop of Q_1 prevents the saturation of Q_4.

To realize negative input weights one can make use of the circuit shown in Fig. (4). This is basically the Wilson's current source [7] employed in a novel manner with inputs at both A and B. From Fig. (4) it follows that the A input will furnish a positive weight while the B input will have an associated negative weight. Note that the (incremental) input resistance at B is β times less than r_e of Q_2 when operating at a bias current of (I_1-I_2). It should be also noted that the output resistance at C is approximately β times higher than the output resistance of a common-emitter transistor, r_o.

As a further design innovation Fig. (5) presents a novel multiple-output current-mirror where the biasing voltage V_B is provided from a suitable low impedance point. For this mirror the current gain $\frac{I_o}{I_i} \simeq 1 - \frac{2}{\beta}$ independent of the number of outputs. The input resistance is $\simeq 2n\, V_T/I_i$ where $n \simeq 1.1$ and $V_T \simeq 26$ m volts. An output resistance of approximately $(1 + \beta)r_o$ is also achieved.

Dynamic response

Since the circuit of Fig. (1) operates on a current-mode principle a high operating speed should be expected. This has been verified both experimentally and using computer simulation routines. The experiments were carried on prototype gates constructed using the RCA transistor array CA 3046 ($f_t \simeq$ 550 MHz) for the NPN devices and the Analog Devices AD 821 ($f_t \simeq$ 200 MHz) IC pairs for the PNP devices. For computer simulation the programs SINC [10] was employed. This is a nonlinear transient analysis program employing a modified Ebers-Moll model for the bipolar transistor. Analysis was performed on a 5-valued gate much like the one shown in Fig.(1) but implementing both E(x) and N(x) simultaneously. The input excitation was a 4 ma current pulse of 50 ns duration and with 15 ns rise and fall times. The output current signals monitored across 10Ω resistors in the collector leads of the output PNP transistors are shown in Fig. (6). Note that the output rise and fall times are approximately 30 ns while the delay time is ≃ 15 ns. Also, the N(x) output exhibits an overshoot of about 12%. This is caused by the current charging the stray capacitances between the substrate (ground) and the collector junctions of Q_4, Q_5, Q_6 and Q_7. The overshoot becomes negligible if the input pulse has rise and fall times of ≃30 ns. Finally, we note that excellent agreement with the computer-predicted results were obtained experimentally.

IV. AN ALL-NPN CURRENT-MODE MT(R) GATE

The possibility of the fabrication of high quality complementary transistorson the same monolithic chip was reported recently [8],[9]. However, such fabrication at present is considerably more expensive than the fabrication of IC's with only NPN transistors. To avoid this drawback an all-NPN-transistors version of a current-mode MT(R) gate has been developed. One example of such circuits is shown in Fig.(7).

The operation of this circuit is as follows. Current signal eI_o is applied to the emitter of the transistor Q_1. The common-base connected transistor Q_1 provides low input impedance for the circuit, and effectively isolates node S from the parasitic capacitances present at the input. The voltage signal obtained at node S is translated to the base of the emitter-follower Q_7. The voltage at the emitter of Q_7 is compared with the threshold voltages connectd to the differential pairs Q_{19} - $Q_{20}, \ldots,$ Q_{25} - Q_{26}.

The collectors of the differential pairs serve directly as the output terminals of the gate. Depending on the transfer characteristic to be realized the appropriate collectors are grouped together and connected to the input terminal of the succeeding gate in the logic network. In this particular example the output signal is monitored as a voltage drop across resistor R_5.

The static transfer characteristic of this gate at different temperatures was simulated with the program SINC. Results of the simulation reveal the excellent temperature compensation inherent in the design. The relative drift between the emitter voltage of Q_7 and the threshold voltages at the emitters of $Q_{28},\ldots,Q_{31}$ is $\simeq -1.5$ mV/$^{\circ}$C. The temperature coefficent of the output current is $\simeq + 0.08\%^{\circ}$C. This temperature compensation can still be improved if the biasing currents for transistors Q_3 and Q_8 are obtained by using appropriate resistors R_3' and R_4' connected to V_{CC}=0.V. Due to space limitation we show in Fig. (8) only the transfer characteristic computed at 300^{o}K.

The time response of the gate was also simulated and the result is shown in Fig.(9). When a pulse with rise and fall times of 15 ns is applied at the input, an output signal with rise and fall time of $\simeq$ 30ns is obtained. We note that this response is slightly slower than the one shown in Fig.(6). The reason for this is the time constant $\tau=R_2\,C_s$, where C_s is the stray capacitance at the base of Q_7 in Fig. (7). The response can be speeded-up easily if smaller value is chosen for R_2 and the current through R_2 is increased accordingly.

V. FIVE-VALUED STORAGE ELEMENT

As an example of possible application of the MT(R) gates we present a 5-stable circuit which is conceptually similar to a bistable. The logic diagram and the circuit diagram of a set-reset 5-stable are shown in Fig. (10)(a) and (b). In this realization feedback is applied around an E element realizing a stair-case transfer characteristic with five logic levels. Other logic elements utilized are multi-valued "AND" and "OR" elements commonly referred to as min and max gates, respectively. The logic equation describing this circuit is

$$Y = (x \cdot c + y) \cdot (x + \bar{c}),$$

where x is the input variable, y is the present output of the E element, c, $\bar{c}$ are clock pulses and Y is the next state. The electronic circuit implementation of this network is based on a voltage-mode principle where the logic values are represented by the voltage levels: "0" ↔ -6.3V, "1" ↔ -5.6V, "2" ↔ -4.9V, "3" ↔ -4.2V and "4" ↔ -3.5V. A "diode-logic" principle is used to implement the operations min and max. For example, transistors Q_7-Q_8 realize the max operation $(x+\bar{c})$ at their common emitter. The transfer characteristic E(x) is realized by the current switching in the differential pairs Q_{34} - $Q_{35},\ldots,$ Q_{40} - Q_{41}. The outputs y and $\bar{y}$ are available at the outputs of the voltage level shifters.

Complex sequential functions can be synthesized with such circuits. For instance a cascade of two of the above circuits realize a 5-stable master-slave. A feedback connected shift-register which generates pseudorandom numbers [6] has been constructed of such modules. The prototype instrument is performing satisfactorily at frequencies up to 3 MHz.

VI. CONCLUSIONS

Practical circuit implementations for MT(R) gates have been developed, computer simulated and experimentally investigated. Also the design of R-storage elements has been demonstrated. The circuits given are suitable for production using IC technology. The MT(R) elements enable flexibility of the design of R-valued switching systems by offering great variety of logic modules. Applications of MT(R) logic in the area of arithmetic networks to the design of adders, base conversion networks and parallel multipliers has been successfully demonstrated [4].

REFERENCES

[1] D. Hampel, L. Micheel and K. Prost,"Threshold Logic Multiplier", NAECON Record's, pp. 288-295,

[2] D.Hampel, "Multi-function Threshold Gate", IEEE Trans. Comp., Vol.C-22, pp.197-203, February, 1973.

[3] D.R. Haring, "Multi-Threshold Threshold Elements", IEEE Trans. on Computers, Vol.EC-15, No.1, pp.45-65, February, 1966.

[4] A. Druzeta, "Multi-Valued Multi-Threshold Logic", Ph.D. Thesis, Univ. of Toronto, in preparation.

[5] C. Lam and Z.G. Vranesic, "Multiple-Valued Pseudorandom Number Generator", Proc. of the 1974 International Symposium on Multiple-Valued Logic, Morgantown, W.V., pp. 45-58, May, 1974.

[6] John Newton, "A Pseudorandom Sequence Generator Using Quinary Logic", B.A.Sc. Thesis, Univ. of Toronto, March, 1974.

[7] G.R. Wilson, "A Monolithic Junction-FET-n-p-n Operational Amplifier", IEEE J. Solid State Circuits, Vol. SC-3, No.4, pp.341-347, December, 1968.

[8] S.C. Su and J.D. Meindl, "A New Complementary Monolithic Transistor Structure", IEEE J. Solid-State Circuits, Vol. SC-7, No.2, pp.170-171, April, 1972.

[9] R.A. Stehlin and G.W. Niemann, "Complementary Transistor-Transistor Logic (CT^2L)- An Approach to High-Speed Micropower Logic", IEEE J. Solid-State Circuits, Vol. SC-7, No.2, pp.153-160, April, 1972.

[10] Available from the Computer Research Lab., University of Calif., Berkeley.

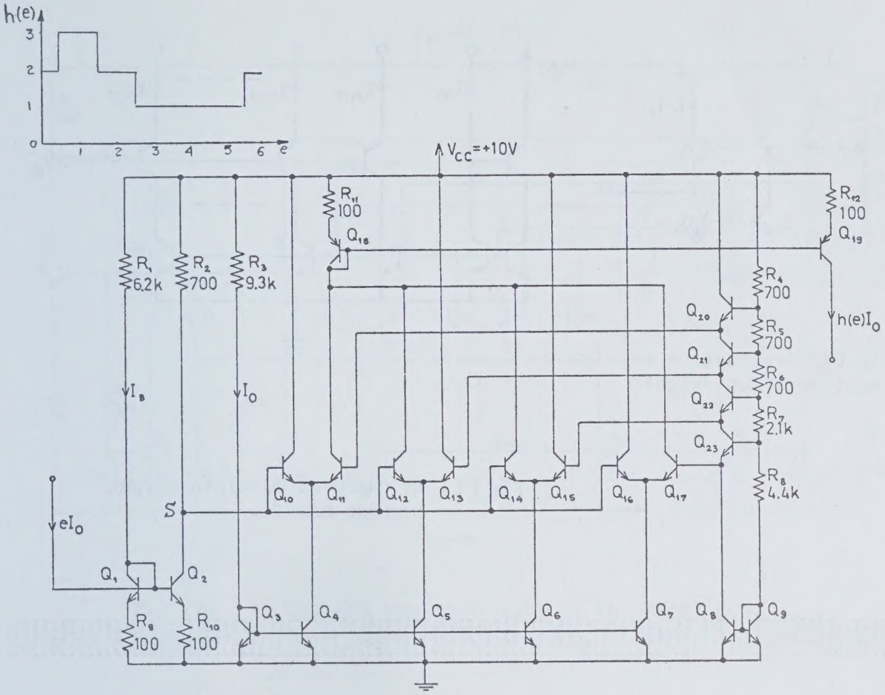
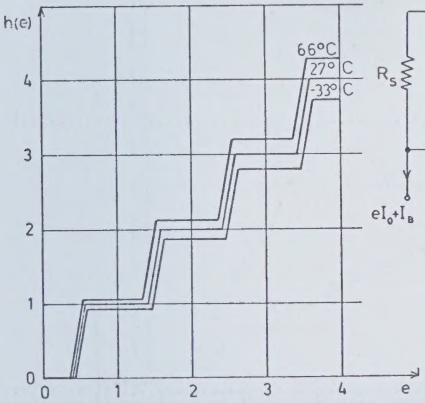

h(e)
3
2
1
0
1
2
3
4
5
6
e
Vcc=+10V
R1 6.2k
R2 700
R3 9.3k
R11 100
Q18
R12 100
Q19
R4 700
R5 700
R6 700
R7 2.1k
R8 4.4k
Q20
Q21
Q22
Q23
h(e)I0
IB
I0
eI0
S
Q10
Q11
Q12
Q13
Q14
Q15
Q16
Q17
Q1
Q2
R9 100
R10 100
Q3
Q4
Q5
Q6
Q7
Q8
Q9

Fig.(1) A transfer characteristic h(e) and its implementation



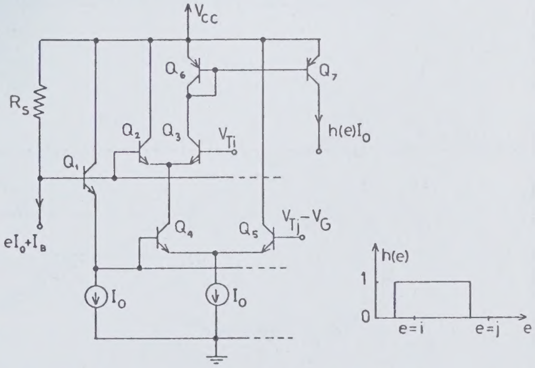

Vcc
Q6
Q7
RS
Q2
Q3
VTi
h(e)I0
Q1
eI0+IB
Q4
Q5
VTj-VG
I0
I0
h(e)
1
0
e=i
e=j
e

Fig. (3) Illustrating the concept of stacking differential pairs to realize arbitrary h(e).

Fig.(2) Measured temperature dependence of the static characteristics of an MT(R) gate such as the one in Fig.(1) but implementing a staircase function.

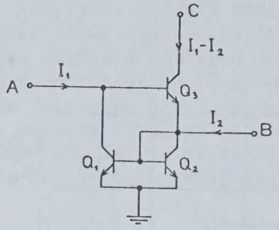

C
I_1-I_2
A
I_1
Q_3
I_2
B
Q_1
Q_2

Fig.(4) Realization of negative input weights

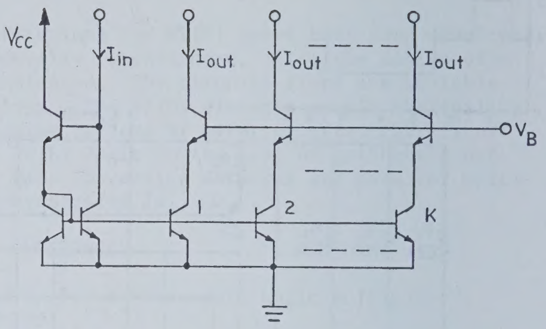

V_{CC}
I_{in}
I_{out}
I_{out}
I_{out}
V_B
1
2
K

Fig.(5) An improved multiple-output current mirror.

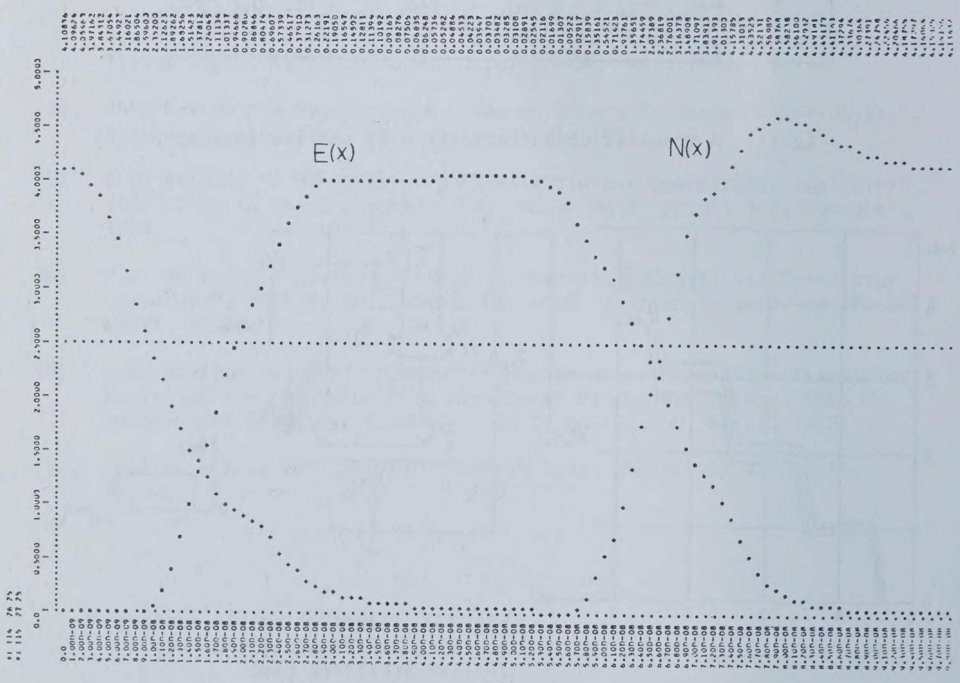


Fig.(6) Computer-calculated dynamic response of the MT(R) gate of Fig. (1) but implementing E(x) and N(x) functions.

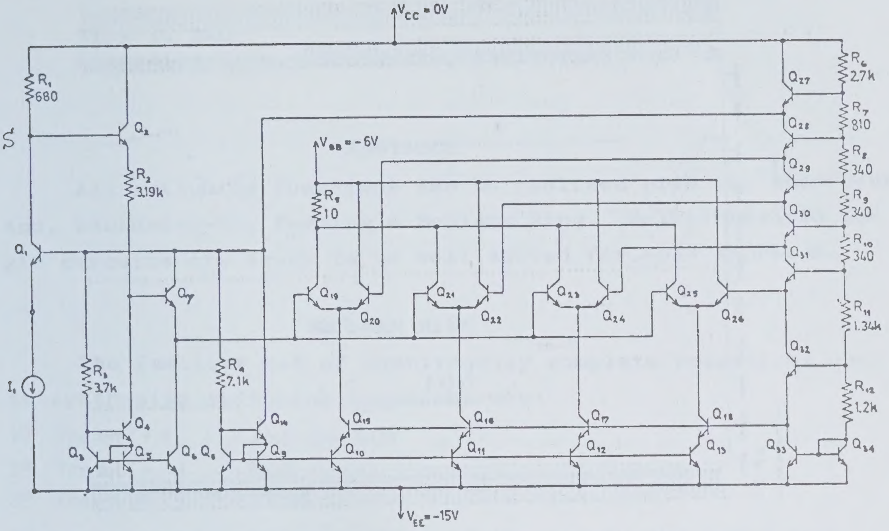

V_{CC} = 0V
R_1
680
Q_2
R_2
3.19k
Q_1
I_1
R_3
3.7k
Q_3
Q_4
Q_5
Q_7
Q_6
Q_8
R_4
7.1k
Q_9
Q_{14}
V_{BB} = -6V
R_5
10
Q_{19}
Q_{20}
Q_{21}
Q_{22}
Q_{23}
Q_{24}
Q_{25}
Q_{26}
Q_{15}
Q_{10}
Q_{16}
Q_{11}
Q_{17}
Q_{12}
Q_{18}
Q_{13}
V_{EE} = -15V
R_6
2.7k
Q_{27}
R_7
810
Q_{28}
R_8
340
Q_{29}
R_9
340
Q_{30}
R_{10}
340
Q_{31}
R_{11}
1.34k
Q_{32}
R_{12}
1.2k
Q_{33}
Q_{34}

Fig. (7) An all-NPN implementation of the MT(R) gate

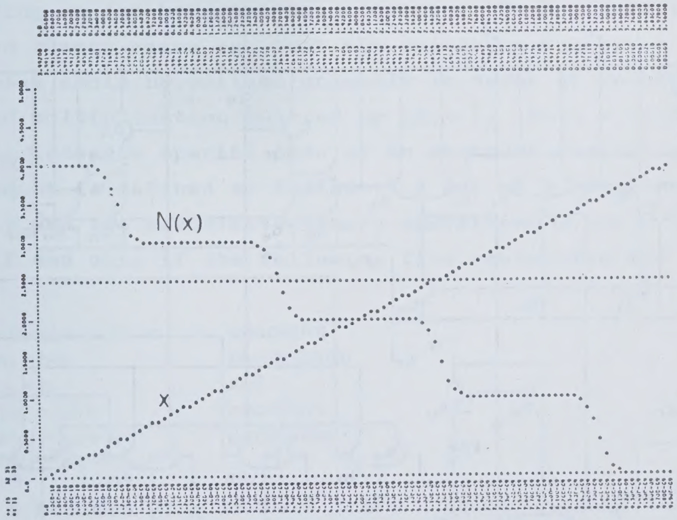


Fig. (8) The transfer characteristic of the gate in Fig. (7) at 300^{O}K obtained by computer simulation.

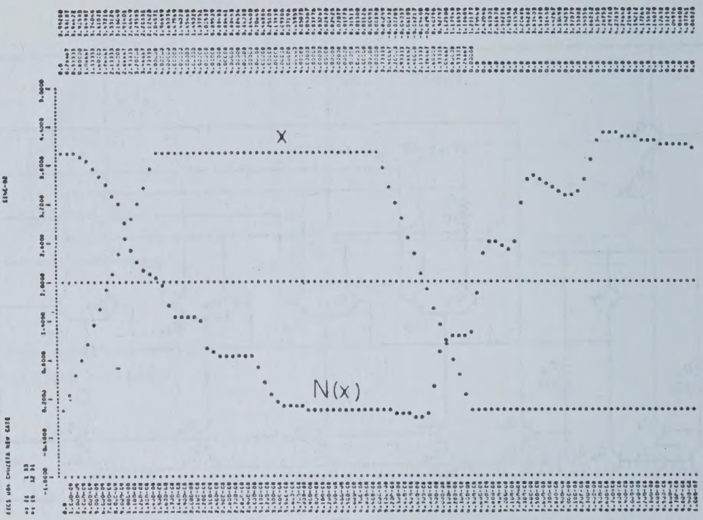


Fig. (9) Computer simulation results of the dynamic response of the gate in Fig. (7).

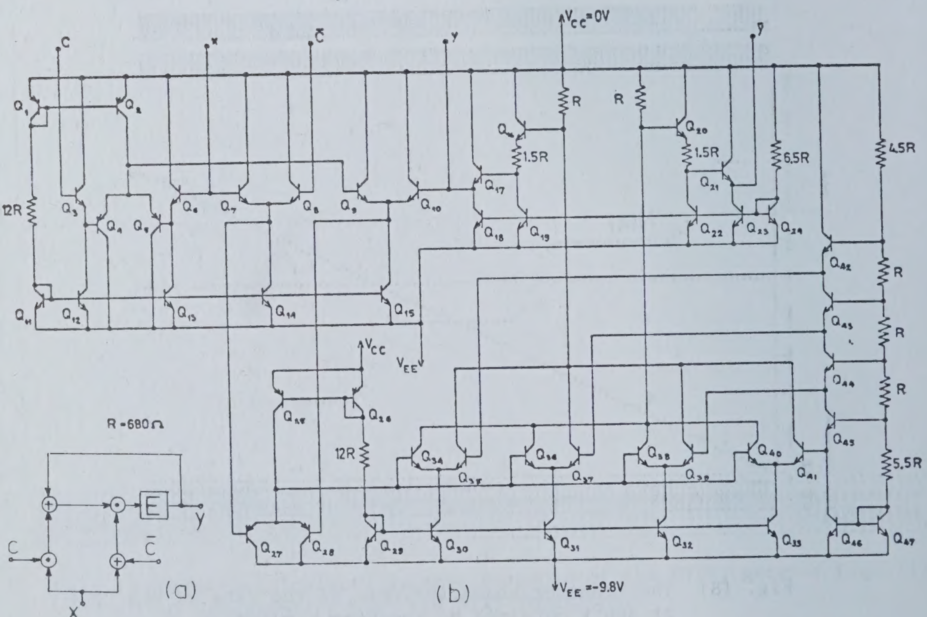

C
x
C̄
y
$V_{CC}=0V$
ȳ
Q_1
Q_2
Q_3
Q_4
Q_5
Q_6
Q_7
Q_8
Q_9
Q_{10}
12R
Q_{11}
Q_{12}
Q_{13}
Q_{14}
Q_{15}
Q_{16}
1.5R
Q_{17}
Q_{18}
Q_{19}
R
R
Q_{20}
1.5R
Q_{21}
6.5R
Q_{22}
Q_{23}
Q_{24}
4.5R
Q_{42}
R
Q_{43}
R
Q_{44}
R
Q_{45}
5.5R
V_{CC}
V_{EE}
Q_{25}
Q_{26}
12R
R = 680Ω
Q_{34}
Q_{35}
Q_{36}
Q_{37}
Q_{38}
Q_{39}
Q_{40}
Q_{41}
Q_{27}
Q_{28}
Q_{29}
Q_{30}
Q_{31}
Q_{32}
Q_{33}
Q_{46}
Q_{47}
$V_{EE}=-9.8V$
E
y
C
C̄
X
(a)
(b)

Fig. (10) (a) The logic diagram and (b) the circuit diagram of a 5-stable element.

MULTITHRESHOLD LOGIC AND BOOLEAN RING SYNTHESIS

TICH T. DAO
Signetics Corp., Sunnyvale, California

ABSTRACT

All switching functions can be realized with two operators And, Exclusive-Or, forming a Boolean Ring. Multithreshold logic circuits are shown to be well suited for this approach.

BOOLEAN RING

The familiar set of functionally complete operations used to synthesize switching functions are:

1° The set $(+, \bullet, ')$ AND, OR, NOT
2° The set $(+, ')$ NOR
3° The set $(\bullet, ')$ NAND

Electronic gate implementation of the second and third sets correspond to the NOR and NAND logic. Optimum synthesis of switching functions usually begins with the Shannon Canonical expansion in sum of product terms and the determination of an optimum set of prime implicant according to some criteria depending on the hardware requirements. Reed and Muller (1) noticed twenty years ago that any switching functions of n variables could be written uniquely in terms of Modulo 2 addition and multiplication denoted by $(\oplus, \bullet)$. Such a representation is indeed a special case of an abstract algebra called Ring which is defined as follows:" A set of R elements (a,b, c,....) and two associated binary operations $\oplus$ and $\bullet$ form a Ring if and only if the following five postulates are satisfied:

(R_1) $a\oplus(b\oplus c) = (a\oplus b)\oplus c$ associative
(R_2) $a\oplus b = b\oplus a$ commutative
(R_3) $a\oplus x = b$ $x \epsilon R$
(R_4) $a(bc) = (ab)c$ associative
(R_5) $a(b\oplus c) = ab \oplus ac$ distributive
$(b\oplus c)a = ba \oplus {}^{a}c$

It is a Boolean Ring if $a \cdot a = a$ (Idempotence)

Also any Boolean Algebra forms a Boolean Ring if $a \oplus b = a'b + ab'$. If a,b are binary then $\oplus$ becomes Modulo 2 addition.

It has been conjectured that the realization of switching function with these two operators is more economical than the conventional form. Minimization of the EX-OR expressions has been attempted but remains an open subject. Without loss of generality consider a function of three variables expanded both Shannon form (1) and generalized Reed-Muller form (2):

Shannon Canonical Expansions

$$f(x,y,z) = f_0x'y'z' + f_1x'y'z + f_2x'yz' + f_3xy'z' + f_4x'yz' + f_5xy'z + f_6xyz' + f_7xyz \qquad (1)$$

Where: $fi = (0,1)$

Generalized Reed-Muller Canonical Expansions (2)

$$f(x,y,z) = a_0 \oplus a_1\dot{x} \oplus a_2\dot{y} \oplus a_3\dot{z} \oplus a_4\dot{x}\,\dot{y} \oplus a_5\dot{x}\,\dot{z} \oplus a_6\dot{y}\,\dot{z} \oplus a_7\dot{x}\dot{y}\dot{z}$$

Where:

$\dot{x}$ = either x or x' $\quad \dot{y}$ = either y or y' $\quad \dot{z}$ = either z or z'

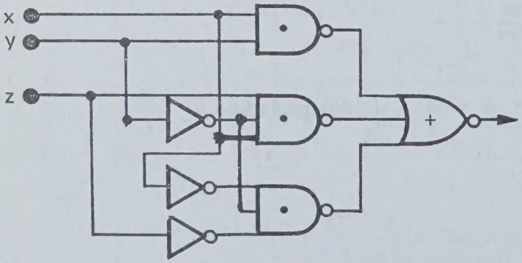
$a_i = (0,1)$

One would need both the primed and unprimed variable in (1), but only one or the other in (2). In the former case, an extra NOT operator is required for each variable, therefore additional circuit and delay. In the latter case, any available polarity of the input variable is directly usable.

Furthermore, if n is the number of input variable then all the product terms are in general of order n, unless some common prime implicants of lower order could be found. In R-M expansion the maximum number of product terms of order $k<n$ is at the most equal to $\binom{n}{k}$, particularly there is only one term of maximum order n.

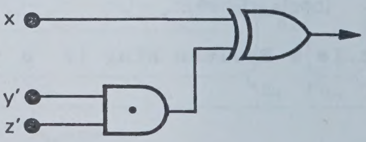
For example, take a following comparative example. Given

$f = xy + xy'z + x'y'z'$ then



Minimal R-M Form :

$f = x \oplus y'z'$



The advent of large scale integration has brought to the forefront the basic idea of using complete array to execute combinational logic function such as the programmable logic array or PLA. Some most important design criteria are regularity, ease of design and testability. It has been pointed out recently by Reddy (2) that Boolean Ring network lends itself to be easily tested. In fact, a network realizing n-variable logic function needs only $4 + \sum_{i=1}^{[\log_2 2t]} \binom{n}{i}$ test set to detect t stucks at faults. ($[x]$) denotes the largest integer less than or equal to x . For example, for t = 1 there are only (n+4) test patterns, independently of the function being realized.

In spite of all the potential advantages of the Boolean Ring, there has been no practical application known at this time. One of the main reason is the lack of simple and high-speed gates, which can perform multiple variables EX-OR operations.

The conventional NAND/NOR implementation of the EX-OR circuit requires at least three levels of logic, NOT and OR, as shown in fig. (1,2). The number of And gate needed to achieve this minimum is $\sum_{1}^{n} \binom{n}{i}$ where n is the number of input.

In general a n input EX-OR is formed by cascading two input EX-OR circuits, forming a binary tree network. The total delay is equal to $\lceil \log_2 n \rceil$.

MULTITHRESHOLD LOGIC

One of the major property of the EX-OR is its total symmetry with respect to the input set. It lends itself to threshold logic implementation with all inputs equally weighed. However, Minsky and Papert (3) have proven that an EX-OR or equivalently a Parity function is not linearly separable, but requires a number of separating planes exactly equal to the number of input variable.

It could also be shown that two levels of threshold gates suffice to implement any EX-OR function. This results from the following property that an EX-OR operator is also a Modulo 2 adder and therefore can be expressed in terms of an arithmetic sum:

Theorem:

$$x_1 \oplus x_2 \oplus \ldots \oplus x_n = \sum_{j=1}^{N} (-2)^{j-1} \sum_{a_1 < a_2 \ldots < a_j} x_{a1}\, x_{a2} \ldots x_{aj}$$

Examples:

1) $x_1 \oplus x_2 = x_1 + x_2 - 2x_1x_2$

2) $x_1 \oplus x_2 \oplus x_3 = x_1 + x_2 + x_3 - 2(x_1x_2 + x_1x_3 + x_2x_3) + 4x_1x_2x_3$

3) $x_1 \oplus x_2 \oplus x_3 \oplus x_4 = x_1 + x_2 + x_3 + x_4 - 2(x_1x_2 + x_1x_3 + x_1x_4 + x_2x_3 + x_2x_4)$
$+ 4(x_1x_2x_3 + x_1x_2x_4 + x_1x_3x_4 + x_2x_3x_4) - 8(x_1x_2x_3x_4)$

The term $\sum_{a_1 < a_2 \ldots < a_j} x_{a1}\, x_{a2} \ldots x_{aj}$ is actually the sum of all possible product terms of order $j \leq n$. The corresponding threshold gate has threshold j. Simplified implementation of the equality in example 3 gives a two-level threshold logic of 4 input EX-OR operator (fig. 3).

We have surveyed known logic implementations of the multi-input EX-OR, either by NAND/NOR gates or by dual level threshold logic gates. We have seen in both cases how the delay and the complexity of the network increase rapidly with increasing input variables.

Multithreshold logic circuits offer a direct and simple approach to the hardware realization. By definition, a k threshold realization of a Boolean function $f(x_1, x_2, \ldots, x_n)$ is an ordered (n+k)-tuples $(w_1, \ldots W_n;\ T_1, \ldots T_k)$ of real numbers, for which the following holds:

$$f(x_1, x_2, \ldots\ldots, x_n) = 1 \text{ if and only if } \left[\begin{array}{ll} \text{weights}: & W = (w_1, w_2, \ldots w_n) \\ \text{thresholds}: & T = (T_1, T_2, \ldots T_k) \\ \text{with} & T_1 < T_2 < T_k \end{array}\right]$$

$$T_{2j-1} \leqslant \sum_i w_i x_i < T_{2j}$$

By referring to the graph in fig. 4, it can be easily seen the EX-OR of n variables is realizable with a multithreshold logic gate whose thresholds are the set of integers: $T_j = j \leqslant n$ if all the weights are chosen equal to unity.

Furthermore, by applying Haring's theorem (4) to this particular case we can state:

when an n-input EX-OR is realized by (1,...1; 1,2,3,..., n) let a threshold function realized by a structure (1,..., 1;i) denoted by f_i, then f may be expanded as :

$$f = f_1 f_2' + f_3 f_4' + \qquad + f_{n-1} f_n' \quad \text{if n is odd}$$
$$= f_1 f_2' + \qquad + f_n \quad \text{if n is even}$$

for n=4 $f = f_1 f_2' + f_3 f_4'$

Fig. 5 shows the block diagram of 4 inputs EX-OR realizing the above relation.

The true inputs are first converted into unit currents which are summed up and converted back into voltage V_A proportional to the number of true inputs. This voltage is then simultanously compared to 4 equally spaced levels by means of four detectors, which are coupled in pairs forming each one a window detector. Their two outputs are mixed in a wired OR function.

The same block diagram can be extended to EX-OR of larger number of inputs. The practical limitation is set by the maximum value of V_A and the minimum tolerable voltage (v) needed for each input. $n \leq V_A/v$

The circuit diagram in fig. 5 represents an experimental ECL version of the double-threshold EX-OR. The two signal current sources are set each by a current mirror scheme. The voltage swing V_A is limited by two clamps. The circuit uses altogether ten transistors not counting the bias circuit which is shared by many. It has a nominal delay of 1,5 ns, dissipates 50 mw and occupies an area of about $60 mil^2$.

The full adder, building block of all binary arithmetic functions, is essentially a triple-input EX-OR circuit, coupled with a Majority function. Multithreshold logic implementation appears to be naturally suited for it. The diagram in fig. 6 shows one possible realization. One of its salient characteristics is that both Carry and Sum are generated at the same time rather than in sequence as in most common circuits.

It must be mentioned finally that several types of multiple inputs EX-OR circuits designed according to the technique presented above have been successfully fabricated and proved to be superior to their conventional counterparts, in terms both of speed and devices count.

CONCLUSION

We have shown that Boolean Ring through generalized Reed-Muller expansion theoretically offers a simple mean to synthesize switching function if high-speed multiple EX-OR hardware were available. It has been proven that multithreshold logic circuits are well adapted to the realization of the needed EX-OR gate, several types have been built and tested with success. The new approach thus offers a new set of tools to the logician who needs not confine himself to synthesize switching functions in terms of NAND/NOR logic, but also in terms of the Boolean Ring, EX-OR, AND, R-M expansions.

REFERENCES

(1) D.E. Muller: Application of Boolean Algebra to Switching Circuit Design and to Error Detection, I.R.E. Transactions-E.C., Sept. 1954, pp. 6-12.

(2) S.M. Reddy: Easily Testable Realizations for Logic Functions, I.R.E. Trans. on Comp. Nov. 1972, pp. 1183-1188.

(3) Minsky and Papert: Perceptrons, M.I.T. Press, 1969.

(4) J. Massey: Threshold Decoding, M.I.T. Press, pp. 119.

(5) D.R. Haring: Multithreshold Threshold Elements, I.E.E. Trans. on Comp., Feb. 1966, pp. 45-65.

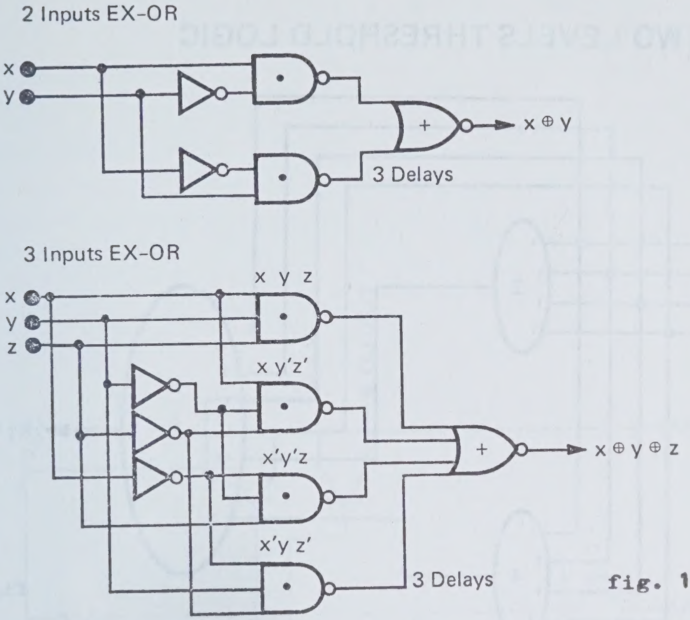

2 Inputs EX-OR
x
y
+
x ⊕ y
3 Delays
3 Inputs EX-OR
x y z
x
y
z
x y'z'
x'y'z
x'y z'
+
x ⊕ y ⊕ z
3 Delays

fig. 1

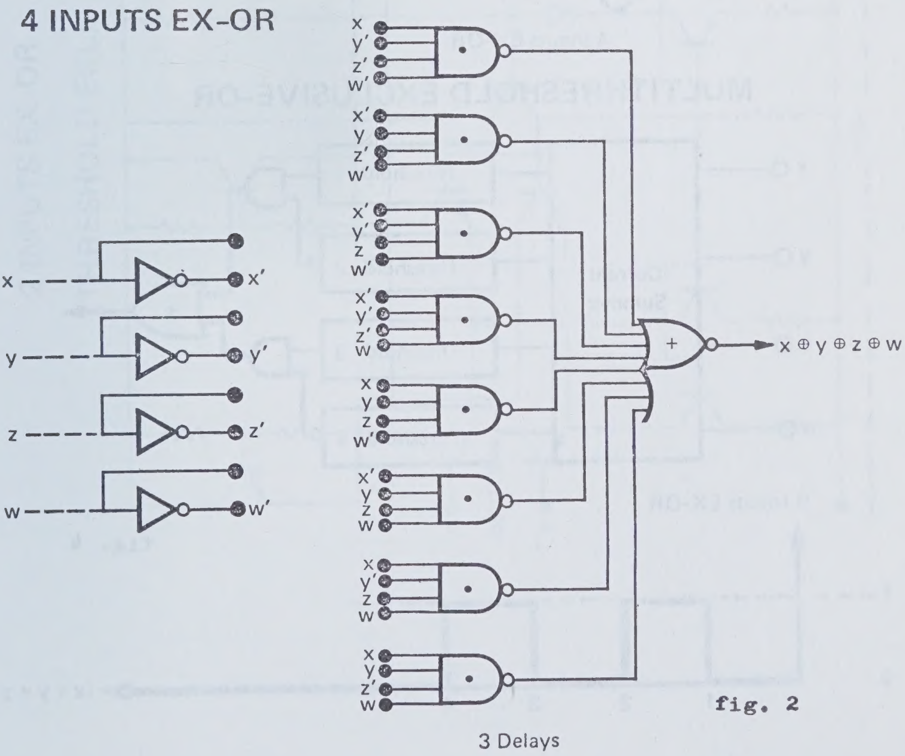
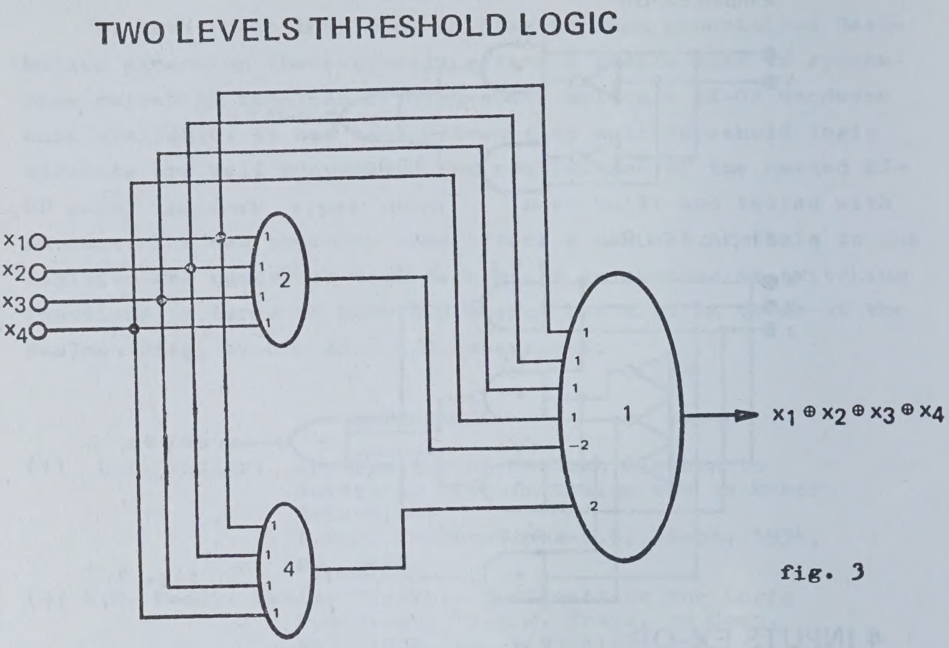

4 INPUTS EX-OR
x
y'
z'
w'
x'
y
z'
w'
x'
y'
z
w'
x'
y'
z'
w
x
y
z
w'
x'
y
z
w
x
y'
z
w
x
y
z'
w
x
x'
y
y'
z
z'
w
w'
+
x ⊕ y ⊕ z ⊕ w
3 Delays

fig. 2


TWO LEVELS THRESHOLD LOGIC
x1
x2
x3
x4
1
1
1
1
2
1
1
1
1
4
1
1
1
1
-2
2
1
$x_1 \oplus x_2 \oplus x_3 \oplus x_4$

4 Inputs EX-OR

fig. 3

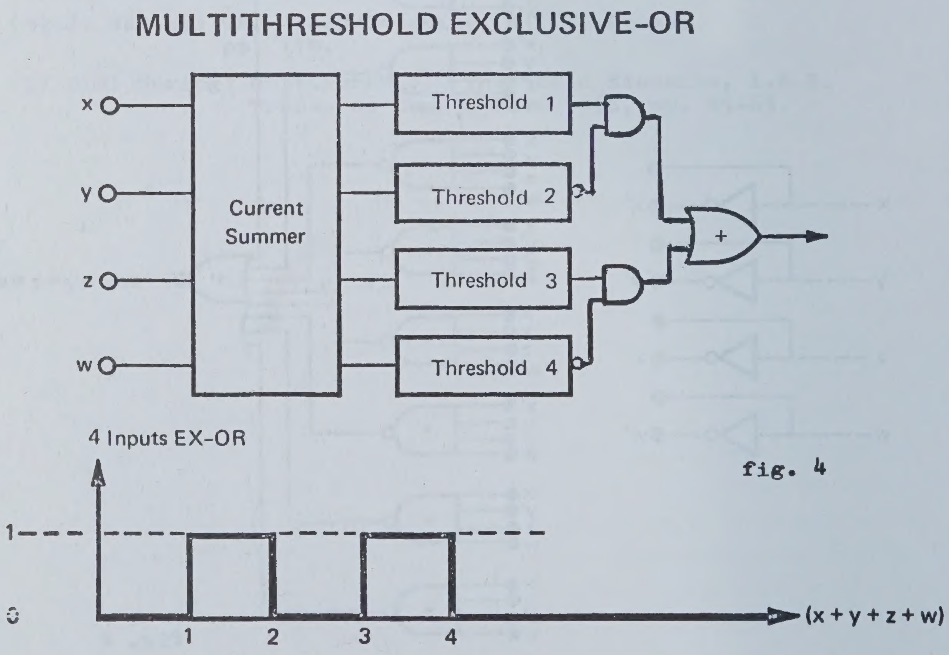
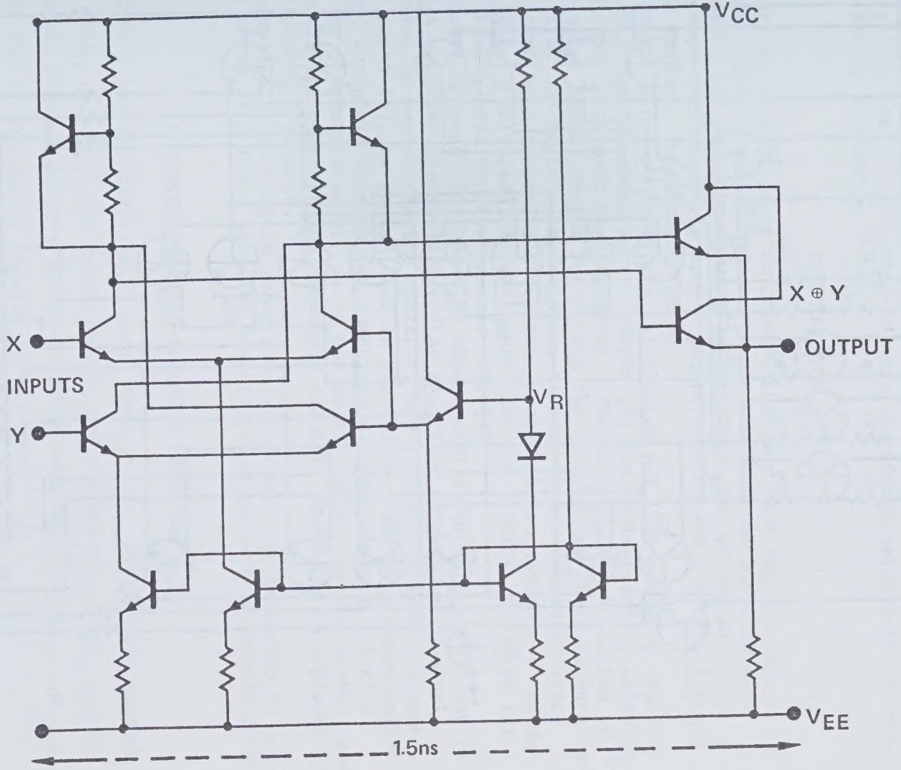

MULTITHRESHOLD EXCLUSIVE-OR
x
y
z
w
Current Summer
Threshold 1
Threshold 2
Threshold 3
Threshold 4
+
4 Inputs EX-OR
1
0
1
2
3
4
(x + y + z + w)

fig. 4

2 INPUTS EX-OR

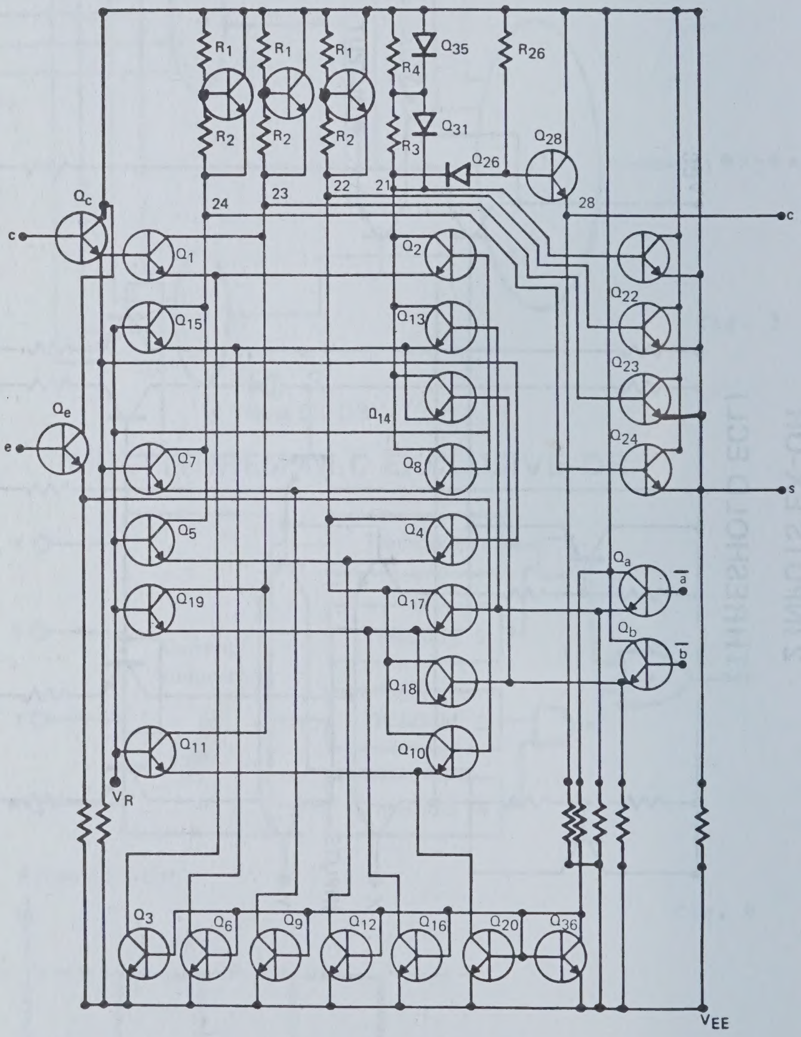
(THRESHOLD ECL)



V_{CC}
$X \oplus Y$
OUTPUT
X
INPUTS
Y
V_R
V_{EE}
1.5ns

fig. 5

THRESHOLD FULL ADDER


R_1
R_1
R_1
R_4
Q_{35}
R_{26}
R_2
R_2
R_2
R_3
Q_{31}
Q_{28}
Q_{26}
24
23
22
21
28
Q_c
c
Q_1
Q_2
c
Q_{22}
Q_{15}
Q_{13}
Q_{23}
Q_e
e
Q_{14}
Q_{24}
Q_7
Q_8
s
Q_5
Q_4
Q_a
$\overline{a}$
Q_{19}
Q_{17}
Q_b
$\overline{b}$
Q_{18}
Q_{11}
Q_{10}
V_R
Q_3
Q_6
Q_9
Q_{12}
Q_{16}
Q_{20}
Q_{36}
V_{EE}

$S = (ab \oplus c \oplus e)$

$C = \text{Majority }(ab, c, e)$

fig. 6

WEIGHTED-MAJORITY-LOGIC DECODING USING NONORTHOGONAL PARITY CHECKS- - AN APPLICATION

W. E. ROBBINS
Department of Computer Science
North Carolina State University
Raleigh, North Carolina 27607

ABSTRACT- It is shown that any decoding function for a linear binary code can be realized as a weighted majority of nonorthogonal parity checks. An example is given of a four-error-correcting code that is neither L-step orthogonalizable nor one-step majority decodable using nonorthogonal parity checks and yet is one-step weighted-majority decodable using only ten nonorthogonal parity checks.

INTRODUCTION

The purpose of this paper is to show that any decoding function for a linear binary (block code or convolution code with fixed decoding constraint length) can be realized as a weighted majority of nonorthogonal parity checks. This result is both a generalization and a specialization of previous results, as indicated below.

In a recent correspondence, Ng [1] pointed out that one-step majority decoding using nonorthogonal parity checks [2] can be improved by allowing the zero parity check more than one vote. The natural generalization of this is to allow parity checks other than the zero parity check to have extra votes; i.e., to replace the majority element by a weighted-majority element. It turns out that this further generalization yields a decoding algorithm that is universal, in the sense that any decoding function for any linear binary code can be realized [3], [8].

In a previous paper [4], Rudolph showed that every linear code can be decoded by a one-step threshold decoding algorithm using nonorthogonal parity checks, an exponentiation operator, and a linear threshold element. In a later paper [5], he stated that in the binary case the exponentiation operator and threshold element can sometimes be replaced by a majority element. The proper statement, as is now shown, is that in the binary case the exponentiation operator and threshold element can always be replaced by a weighted-majority element.

For simplicity, only block codes are considered; the extension to convolutional codes with fixed decoding constraint length follows easily from [5].

ONE-STEP WEIGHTED-MAJORITY DECODING

Let C be a linear binary (n,k) code, $c=(c_0,\dots,c_{n-1})$ the transmitted codeword, $e=(e_0,\dots,e_{n-1})$ the error vector, and $r=c \oplus e=(r_0,\dots,r_{n-1})$ the received word, where $\oplus$ denotes modulo-2 vector addition. $S=\{s_0,\dots,s_{2^{n-k}-1}\}$ denotes the set of all parity checks

$$s_i = \bigoplus_{j=0}^{n-1} h_{ij} r_j ,$$

where $(h_{i0},\dots,h_{i(n-1)})$ is the ith word in the dual code, and $\oplus$ denotes modulo-2 sum.

One-Step Majority Decoding Rule: Let $x_1,\ldots,x_m$ be Boolean variables with m odd. The majority function is defined to be

$$\text{maj}\ \{x_i,\ldots,x_m\} = \begin{cases} 1, & \text{if } \sum_{i=1}^{m} x_i > \frac{m}{2} \\ 0, & \text{if } \sum_{i=1}^{m} x_i < \frac{m}{2} \ . \end{cases}$$

Then a one-step majority decoding rule for the error in the ith position of the received word is

$$\hat{e}_i = \text{maj}\ \{s_{i_1},\ldots,s_{i_m}\}$$

for m odd and $s_{i_j} \varepsilon S$. The parity checks s_{i_j} are distinct but not necessarily orthogonal.

One-Step Weighted-Majority Decoding Rule: Generalizing the concept of majority function, a weighted-majority function of Boolean variables $x_1,\ldots,x_m$ is defined to be

$$\text{wtd maj}\ \{w_1x_1,\ldots,w_mx_m\} = \begin{cases} 1, & \text{if } \sum_{i=1}^{m} w_ix_i > \frac{1}{2}\sum_{i=1}^{m} w_i \\ 0, & \text{if } \sum_{i=1}^{m} w_ix_i < \frac{1}{2}\sum_{i=1}^{m} w_i \ , \end{cases}$$

where the w_i are positive integers and $\sum_{i=1}^{m} w_i$ is odd. Then a one-step weighted-majority decoding rule for the error in the ith position of the received word is

$$\hat{e}_i = \text{wtd maj}\ \{w_{i_1}s_{i_1},\ldots,w_{i_m}s_{i_m}\} \ ,$$

where the w_{i_j} are positive integers, $s_{i_j} \varepsilon S$ and $\sum_{j=1}^{m} w_{i_j}$ is odd.

Note that a weighted-majority decoding rule is equivalent to a majority decoding rule with parity checks not required to be distinct.

It is now shown that any decoding function for a linear binary (n,k) code can be realized as a weighted majority of parity checks. The approach is to show that any Boolean function that satisfies a certain condition can be realized as a weighted majority of modulo-2 sums of its arguments and then to show that any decoding function satisfies this condition.

Lemma 1:

$$\text{wtd maj } \{w_{i_1} s_{i_1}, \ldots, w_{i_m} s_{i_m}\}$$

$$= \log_{-1} \operatorname{sgn}\left[\sum_{j=1}^{m} w_{i_j} \cos \pi \left(\sum_{j=1}^{n-1} h_{i_j k} r_k\right)\right].$$

where $\log_{-1}$ is the logarithm function to the base -1,

$$\operatorname{sgn}(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{if } x = 0 \\ -1, & \text{if } x < 0 \end{cases}$$

and

$$\log_{-1} \operatorname{sgn}(x) = \begin{cases} 0, & \text{if } x > 0 \\ \text{undefined}, & \text{if } x = 0 \\ 1, & \text{if } x < 0 \, . \end{cases}$$

Proof: Assume wtd maj $\{w_{i_1} s_{i_1}, \ldots, w_{i_m} s_{i_m}\} = 0$. Then

$$\sum_{j=1}^{m} w_{i_j} s_{i_j} < \frac{1}{2} \sum_{j=1}^{m} w_{i_j}$$

or

$$0 < \sum_{j=1}^{m} w_{i_j} (1 - 2s_{i_j})$$

$$= \sum_{j=1}^{m} w_{i_j} \cos \pi s_{i_j}$$

$$= \sum_{j=1}^{m} w_{i_j} \cos \pi \left(\bigoplus_{k=0}^{n-1} h_{i_j k} r_k\right)$$

$$= \sum_{j=1}^{m} w_{i_j} \cos \pi \left(\sum_{k=0}^{n-1} h_{i_j k} r_k\right),$$

where we have used the facts that for Boolean variables x_i,

$$\cos \pi x_i = 1 - 2x_i \quad \cos \pi \left(\bigoplus_{i} x_i\right) = \cos \pi \left(\sum_{i} x_i\right).$$

Then

$$\log_{-1} \operatorname{sgn}\left[\sum_{j=1}^{m} w_{i_j} \cos \pi \left(\sum_{k=0}^{n-1} h_{i_j k} r_k\right)\right] = \log_{-1} (1) = 0.$$

An analogous argument holds for the case

$$\text{wtd maj } \{w_{i_1} s_{i_1}, \ldots, w_{i_m} s_{i_m}\} = 1 . \qquad \text{Q.E.D.}$$

It is convenient at this point to adopt the following convention. Let $\bar{i}$ denote the binary representation of the number i. If f is a Boolean function, then $f(\bar{i})$ is the value of f at the vertex of the unit cube with coordinates corresponding to the binary representation of i.

Lemma 2: A Boolean function $f(\bar{i})$ can be realized as the $\log_{-1}$ sgn of a finite Fourier cosine series with positive integer coefficients w_{i_j}, where $\sum_{j=1}^{m} w_{i_j}$ is odd, if and only if $f(\bar{0}) = 0$.

Proof: Let $L = \{L_0, L_1, \ldots, L_{2^t-1}\}$ be the complete set of 2^t linear function of t Boolean variables ordered so that $L_j(\bar{k}) = \bar{j} \cdot \bar{k}$, where the dot denotes inner produce over GF(2). Suppose that f can be realized as $\log_{-1}$ sgn of a finite Fourier cosine series

$$\sum_{j=1}^{m} w_{i_j} \cos \pi(L_{i_j}(\bar{k})), \qquad k = 0,1,\ldots,2^{t-1},$$

when the w_{i_j} are positive integers and $\sum_{j=1}^{m} w_{i_j}$ is odd. Then

$$f(\bar{0}) = \log_{-1} \operatorname{sgn}\left(\sum_{j=1}^{m} w_{i_j} \cos_{\pi}(L_{i_j}(\bar{0}))\right)$$

$$= \log_{-1} \operatorname{sgn}\left(\sum_{j=1}^{m} w_{i_j}\right)$$

$$= \log_{-1}(1)$$

$$= 0 .$$

Conversely, suppose that $f(0) = 0$. We show that f can be written as the $\log_{-1}$ sgn of a finite Fourier cosine series with positive integer coefficients with an odd sum, by constructing such a series. Define a set of nonnegative integer Fourier coefficients by

$$W_k = \frac{1}{2} \sum_{i=0}^{2^t-1} f(\bar{i})(1 - \cos \pi L_i(\bar{k})) + \delta_{0k}, \qquad k = 0,1,\ldots,2^t - 1,$$

where

$$\delta_{ij} = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j . \end{cases}$$

The $\log_{-1}$ sgn of the cosine series with these coefficients is a function

$$g(\overline{m}) = \log_{-1} \operatorname{sgn}\left\{ \sum_{k=0}^{2^t-1} w_k \cos \pi L_k(\overline{m}) \right\}$$

$$= \log_{-1} \operatorname{sgn}\left\{ \sum_{k=0}^{2^t-1} \left(\frac{1}{2} \sum_{i=0}^{2^t-1} f(\overline{i})(1 - \cos \pi L_i(\overline{k})) + \delta_{0k} \right) \right.$$

$$\left. \cdot \cos \pi L_k(\overline{m}) \right\}$$

$$= \log_{-1} \operatorname{sgn}\left\{ \frac{1}{2} \sum_{i=0}^{2^t-1} f(\overline{i}) \sum_{k=0}^{2^t-1} \cos \pi L_k(\overline{m}) \right.$$

$$- \frac{1}{2} \sum_{i=0}^{2^t-1} f(\overline{i}) \sum_{k=0}^{2^t-1} \cos \pi L_i(\overline{k}) \cos \pi L_k(\overline{m})$$

$$\left. + \sum_{k=0}^{2^t-1} \delta_{0k} \cos \pi L_k(\overline{m}) \right\}.$$

Noting that $L_k(\overline{m}) = L_m(\overline{k})$ and substituting in the preceding equation gives

$$g(\overline{m}) = \log_{-1} \operatorname{sgn}\left\{ \frac{1}{2} \sum_{i=0}^{2^t-1} f(\overline{i}) \sum_{k=0}^{2^t-1} \cos \pi L_m(\overline{k}) \right.$$

$$- \frac{1}{2} \sum_{i=0}^{2^t-1} f(\overline{i}) \sum_{k=0}^{2^t-1} \cos \pi L_i(\overline{k}) \cos \pi L_m(\overline{k})$$

$$\left. + \sum_{k=0}^{2^t-1} \delta_{0k} \cos \pi L_m(\overline{k}) \right\}.$$

By the orthogonality properties of the cosine functions we see that the first term is zero unless $m = 0$, the only nonzero term in the second sum is that for which $i = m$, and the third term is a constant. Specifically, if we evaluate the nonzero terms,

$$g(\overline{m}) = \log_{-1} \operatorname{sgn}\left(\delta_{0m} \sum_{i=0}^{2^t-1} f(\overline{i}) 2^{t-1} = f(\overline{m}) 2^{t-1} + 1 \right).$$

Then

$$g(\overline{m}) = \begin{cases} 0, & \text{if } m = 0 \\ f(\overline{m}), & \text{if } m > 0 \end{cases}$$

from which we see that $g = f$ if $f(\overline{0}) = 0$. It is easily verified that $\sum_{k=0}^{2^t-1} w_k$ is odd; it then follows that

$$f(\overline{m}) = \log_{-1} \operatorname{sgn}\left(\sum_{w_k>0} w_k \cos \pi L_{,} (\overline{m}) \right)$$

$$= \log_{-1} \operatorname{sgn}\left(\sum_{j=1}^{m} w_{i_j} \cos \pi L_{i_j} (\overline{m}) \right),$$

where m is the number of $w_k > 0$. Q.E.D.

Theorem: Any decoding function for a linear binary (n,k) code can be realized as a weighted majority of nonorthogonal parity checks.

Proof: Let $t_1,\dots,t_{n-k}$ be a basis, over GF(2), for the set S of all parity checks. Then the decoding function $\hat{e}_i$ is a Boolean function of the Boolean variables $t_1,\dots,t_{n-k}$. A basic requirement for any decoding function is that it give the correct estimate in the absence of noise. It then follows from the linearity of C that $\hat{e}_i(\overline{0}) = 0$ and hence by

Lemma 2

$$\hat{e}_i(t_1,\dots,t_{n-k}) = \log_{-1} \operatorname{sgn}\left(\sum_{j=1}^{m} w_{i_j} \cos \pi L_{i_j} (t_1,\dots,t_{n-k}) \right)$$

$$= \log_{-1} \operatorname{sgn}\left(\sum_{j=1}^{m} w_{i_j} \cos \pi S_{i_j} \right),$$

where the w_{i_j} are positive integers and $\sum_{j=1}^{m} w_{i_j}$ is odd. But then by

Lemma 1

$$\hat{e}_i = \text{wtd maj } \{w_{i_1} s_{i_1}, \dots, w_{i_m} s_{i_m}\}.$$ Q.E.D.

EXAMPLE

$$H = \left[\begin{array}{ccccc|c} 1 & 1 & 0 & 0 & 0 & \\ 0 & 1 & 1 & 0 & 0 & \\ 0 & 0 & 1 & 1 & 0 & \\ 0 & 0 & 0 & 1 & 1 & \\ 1 & 1 & 0 & 0 & 1 & \\ 1 & 0 & 1 & 0 & 0 & \\ 0 & 1 & 0 & 1 & 0 & \\ 0 & 0 & 1 & 0 & 1 & \\ 1 & 1 & 0 & 1 & 0 & I_{16} \\ 0 & 1 & 1 & 0 & 1 & \\ 1 & 1 & 1 & 1 & 0 & \\ 0 & 1 & 1 & 1 & 1 & \\ 1 & 1 & 1 & 1 & 1 & \\ 1 & 0 & 1 & 1 & 1 & \\ 1 & 0 & 0 & 1 & 1 & \\ 1 & 0 & 0 & 0 & 1 & \end{array} \right]$$

specifies a (21,5) four-error-correcting binary cyclic code. Duc [6] has verified that this code is neither L-step orthogonalizable [7] nor one-step majority decodable using nonorthogonal parity checks. However, the following set of ten vectors in the row space of H, weighted as indicated, will produce the correct estimate of the error bit in the 0th position of a received word, for any pattern of four or fewer errors, by 9-out-of-17 majority vote.

Vector	Votes
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	2
1 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	2
1 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0	2
1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1	2
1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0	2
1 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0	2
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0	2
1 0 0 1 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0	1
1 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0	1
1 0 0 0 0 0 1 0 0 1 0 0 1 0 0 0 0 0 0 0 0	1

CONCLUSIONS

It has been shown that any decoding function for a linear binary code can be realized as a weighted majority of nonorthogonal parity checks. Furthermore, by example, it has been shown that one-step weighted-majority decoding may be extremely simple, even for codes not decodable by traditional majority decoding rules.

REFERENCES

[1] S. W. Ng, "On Rudolph's majority-logic decoding algorithm," IEEE Trans. Inform. Theory (Corresp.), vol. IT-16, pp. 651-652, Sept. 1970.

[2] L. D. Rudolph, "A class of majority logic decodable codes," IEEE Trans. Theory (Corresp.), vol. IT-13, pp. 305-307, Apr. 1967.

[3] W. E. Robbins, "A group theoretic study of generalized threshold logic,", Ph.D. dissertation, Syst. Inform. Sci., Syracuse Univ., Syracuse, N.Y., June 1971

[4] L. D. Rudolph, "Threshold decoding of cyclic codes," IEEE Trans. Inform. Theory, Vol. IT-15, pp. 414-418, May 1969.

[5] ______, "Generalized threshold decoding of convolutional codes," IEEE Trans. Inform. Theory, vol. IT-16, pp. 739-745, Nov. 1970.

[6] N. Q. Duc, "Pseudostep orthogonalization: A new threshold-decoding algorithm," IEEE Trans. Inform. Theory (Corresp.), vol. IT-17, pp. 766-768, Nov. 1971.

[7] J. L. Massey, Threshold Decoding. Cambridge, Mass.: M.I.T. Press, 1963.

[8] L. D. Rudolph and W. E. Robbins, "One-step-weighted-majority decoding," IEEE Trans. Inform. Theory, vol. IT-18, pp. 446-448, May 1972.

RECENT RESULTS IN LARGE SCALE SYSTEMS

H.H. ROSENBROCK

UMIST, Manchester, England

The canonical decomposition introduced by Kalman shows how redundancy may enter into the way in which a system generates the strictly proper part of its transfer function matrix. A similar redundancy can occur in the generation of the polynomial part of the transfer function matrix, and it leads to a similar fourfold decomposition. Some applications are given to composite systems.

INTRODUCTION

If a system is governed by the equations

$$\left.\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned}\right\} \qquad (1)$$

we can equally well describe it by the system matrix [Rosenbrock, 1970]

$$P(s) = \begin{pmatrix} sI-A & B \\ -C & 0 \end{pmatrix} \qquad (2)$$

As was shown by Kalman [1962] the properties of controllability and observability induce a fourfold decomposition of the system. Three of the four parts so obtained are redundant so far as the input-output behaviour is concerned, being decoupled from input or output or both.

From (2) we obtain

$$G(s) = C(sI-A)^{-1}B \qquad (3)$$

which is strictly proper. More generally we may start from

$$P(s) = \begin{pmatrix} sI-A & B \\ -C & D(s) \end{pmatrix} \qquad (4)$$

where $D(s)$ is a polynomial matrix. From (4) we obtain

$$G(s) = C(sI-A)^{-1}B + D(s) \qquad (5)$$

We then obtain a decomposition into four parts, as before, with respect to the way in which the system generates $C(sI-A)^{-1}B$. This, however, says nothing about the way in which $D(s)$ is generated.

As will be shown, the system can be decomposed into eight parts, four of which correspond to Kalman's decomposition. The other four relate to the way in which D(s) is generated. This might not seem to have much interest, as physical systems are expected to give strictly proper transfer functions. There are, however, a number of physical systems which give a strictly proper G, but which would give a polynomial contribution to G if the corresponding part of the system were not decoupled.

DECOUPLING ZEROS

When we wish to deal with composite systems [Rosenbrock and Pugh, 1974] equation (4) does not provide an adequate description, as the identity of the subsystems is lost. A suitable generalisation is

$$P(s) = \begin{pmatrix} sE-A & B \\ -C & 0 \end{pmatrix} \tag{6}$$

where E is possibly singular, $|sE-A| \not\equiv 0$, and the degree n of $|sE-A|$ is the order of the system. We define a transformation of restricted system equivalence (r.s.e.) by

$$P_1(s) = \begin{pmatrix} M & 0 \\ 0 & I \end{pmatrix} P(s) \begin{pmatrix} N & 0 \\ 0 & I \end{pmatrix} \tag{7}$$

where M,N are real, nonsingular matrices. Then using results due to Kronecker [Gantmacher, 1959] we may bring P(s) by r.s.e. to the form [Rosenbrock, 1974b]

$$P_1(s) = \left(\begin{array}{cc|c} sI-A_1 & 0 & B_1 \\ 0 & I+sJ & B_2 \\ \hline -C_1 & -C_2 & 0 \end{array}\right) \tag{8}$$

where J has all its entries zero except possibly for unit entries in the first superdiagonal. From (8) we obtain

$$G(s) = C_1(sI-A_1)^{-1}B_1 + C_2(I+sJ)^{-1}B_2 \tag{9}$$

in which the first term is strictly proper and the second is polynomial.

In Kronecker's theory, the pencil of matrices

$$(sI-A_1 \quad 0 \quad B_1) \tag{10}$$

may have finite elementary divisors. The zeros of these divisors have been called [Rosenbrock, 1970,1974b] the (finite) input decoupling zeros, and they are the eigenvalues of the uncontrollable part of the system.

Similarly, the pencil

$$(0 \quad I+sJ \quad B_2) \tag{11}$$

will have infinite elementary divisors, and it is natural to look to these for an extension of the idea of controllability, though the usual system-theoretic interpretation of controllability is lost.

In fact it is necessary [Rosenbrock, 1974b] to depart slightly from this direct application of Kronecker's theory. The pencil (11) always has infinite elementary divisors when E is singular, equal in number to the rank defect of E. We therefore define infinite input decoupling zeros as being equal in number to the finite elimentary divisors of

$$(0 \quad sI+J \quad B_2) \tag{12}$$

rather than those of

$$(0 \quad sI+J \quad sB_2) \tag{13}$$

as would follow from a more direct interpretation of Kronecker. With corresponding definitions for infinite o.d. and i.o.d. zeros we can set up a fourfold decomposition of the part of the system giving rise to $C_2(I+sJ)^{-1}B_2$.

With these definitions we can obtain [Rosenbrock, 1974b] the following interesting generalisation of a result due to Kalman [1963].

Theorem 1

Let P,P' be two system matrices having the form (6), neither of which has a (finite or infinite) decoupling zero. Then P,P' are r.s.e. if and only if they give the same $G(s)$.

MINIMAL INDICES

In Kronecker's theory, the pencil

$$(sE-A \quad B) \tag{14}$$

has no minimal index for the rows, because $|sE-A| \not\equiv 0$. It has as many minimal indices for the columns as there are inputs to the system. When G is proper, these minimal indices for the columns are the controllability indices, and their sum is $\nu(G)$, the least order corresponding to G. When G is not proper, the minimal indices are the same [Rosenbrock,1974b] as the invariant dynamical indices of G defined by Forney [1973]. Their sum is the McMillan degree $\delta(G)$ of G, and they are invariant under a generalised transformation [Rosenbrock and Hayton, 1974] of Brunovsky's type.

COMPOUND SYSTEMS

In many branches of systems theory it is important to preserve the identity of subsystems. For example, electrical network theory is concerned with systems made up of subsystems, interconnected according to Kirchoff's laws. The form of P in (6) is sufficiently general to permit such a description [Rosenbrock and Pugh, 1974]. We can then study the way in which the properties of a system depend on the properties of its subsystems and on the method of interconnection.

If a system S is composed of subsystems $S_1, S_2, \ldots, S_N$, and if we distinguish variables by corresponding subscripts, then a system of interconnection corresponding to an analogue computer is

$$\left.\begin{aligned} u_i &= -\sum_{t=1}^{N} F_i^{(t)} y_t + K_i u \\ y &= \sum_{i=1}^{N} L_i y_i \end{aligned}\right\} \qquad (15)$$

One might assume that these equations were general enough to cover all types of linear, time-invariant system, but this is not the case.

Define the complexity c of a system as the sum of the column degrees in its system matrix. Then it can be shown that if S has a strictly proper transfer function matrix, and if P is set up in an appropriate way [Rosenbrock and Pugh, 1974] we have $c = n$. The trivial example shown in the Figure has $c = 2$, $n = 1$, which shows that this circuit cannot be represented as a compound system on an analogue computer. The reason is clear: we can represent each capacitor by an integrator, but we cannot force the outputs of the integrators to equality. This is associated with the fact that analogue integrators are unidirectional devices, whereas a capacitor is not.

The system in the Figure has in fact an infinite decoupling zero. The expression $C_2(I+sJ)^{-1}B_2$ in (9) is therefore zero, even though $(I+sJ)^{-1}$ contains a term in s. The distinction between (15) and Kirchoff's equations which is illustrated by this example suggests that there may be still further interconnection schemes which deserve study.

EXAMPLES

The example in the Figure is trivial, but the same phenomena can occur in a nontrivial way. There are certain rational matrices $G(s)$ which can be realised as the impedance matrix of an RLC n-port, but only if we allow more than $\delta(G)$ reactive elements [Seshu and Reed, 1961]. Sometimes the redundancy which this implies corresponds to uncontrollability and unobservability when the equations are put in state-space form. Sometimes, however, these equations are controllable and observable and the redundancy is attributable to decoupling zeros at infinity [Rosenbrock, 1974a].

As a second example [Rosenbrock, 1974c] the equations of a distillation column can be written, after linearisation around a given working point,

$$\left.\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned}\right\} \qquad (16)$$

where x is a vector of composition changes for each of the components on each of the plates. Because the compositions are expressed as mole fractions, the changes of composition on any plate are related, and we have a set of relations which can be written

$$Qx = 0 \qquad (17)$$

This equation restricts x to a subspace, and because of this fact we can not regard x as a state vector in the usual sense. The equations (16) have in fact a redundancy which corresponds to infinite input decoupling zeros.

REFERENCES

1. FORNEY, G.D., 1973, : Minimal bases of rational vector spaces, with applications to multivariable linear systems, to be published.

2. GANTMACHER, F.R., 1959, : Applications of the theory of matrices, (Interscience).

3. KALMAN, R.E., 1962, : Canonical structure of linear dynamical systems, Proc.Nat.Acad.Sci., Vol. 48, pp. 596-600

4. KALMAN, R.E., 1963, : Mathematical description of linear dynamical systems, J. SIAM Control, Series A, Vol. 1, pp. 152-192.

5. ROSENBROCK, H.H., 1970, : State-space and multivariable theory, (Nelson-Wiley).

6. ROSENBROCK, H.H., 1974a, : Non-minimal LCR multiports, Int.J.Control Vol. 20, pp. 1-16

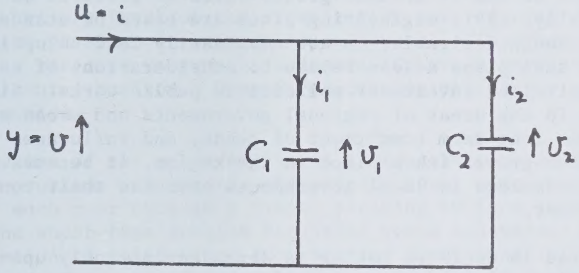
7. ROSENBROCK, H.H., 1974b, : Structural properties of linear dynamical systems, Int.J.Control, Vol.20, pp.191-202

8. ROSENBROCK, H.H., 1974c, : Redundancy in linear, time-invariant, finite-dimensional systems, IFAC, Manchester, to be published.

9. ROSENBROCK, H.H. and HAYTON, G.E., 1974, : Dynamical indices of a transfer function matrix, Int.J.Control,Vol. 20, pp. 177-189

10. ROSENBROCK, H.H. and PUGH, A.C., 1973, : Contributions to a hierarchical theory of systems, Int.J.Control, Vol.19, pp. 845-867

11. SESHU, S. and REED, H.B., 1961, : Linear graphs and electrical networks, (Addison-Wesley).



AN INTEGRATED APPROACH TO LONG RANGE PLANNING FOR URBAN SYSTEMS

H.S. RAO
Systems Control Inc., Palo Alto, California 94304

ABSTRACT

This paper provides an integrated approach to long range planning of urban service systems, including systems such as solid waste, water supply and distribution, sewage disposal, police protection, and flood control. An overall structure for the urban service system is provided in the form of a two-level hierarchy. The first level consists of the decision making authority, namely the city or county council; the second level consists of individual subsystems. The approach described in this paper consists of three types of models: a demand forecasting model, a long range planning model for public facilities and a model for integrating the subsystems. The methodology is computerized and sufficiently general to be useful in other planning studies.

1.0 INTRODUCTION

The planning of urban service systems has hitherto been in response to critical needs or due to assumed growth rates in patterns of land use. Traditionally, civil engineering plans are based on standards or codes which, although desirable, do not necessarily lead to optimal plans. In fact such plans seldom relate to considerations of cost-effective and optimized investment policies in public works. Since decision-making in the areas of regional governments and urban systems planning involves long term commitment of funds, and influences the direction in which growth takes place in the region, it becomes imperative that the decision makers in local governments exercise their controls in an effective manner.

Effectiveness in decision making is dependent not only upon a complete understanding of the organizational structure of the urban service system but also upon three other equally important characteristics of a large scale system. First, there is a need to base the decisions on the latest available information to the decision makers. This information pattern will generally be different for each aspect of the public works system, e.g., water distribution system requires different patterns of information than the solid waste system. Secondly, there is a need to utilize planning tools to develop several alternative plans in response to patterns of growth. An efficient way of developing alternative plans is by building computer based models of each of the subsystems and by using demographic and land use projections to drive the models. The outputs of the models will be long range plans for urban service systems. Thirdly, these plans are evaluated in terms of specific planning criteria and the optimal plan is selected. Usually, it is necessary to make a sequence of decisions over a period of time. Effective decision making now requires an evaluation of the impacts of alternative sequences of decisions.

The objective of this paper is to present an integrated approach to long range planning of urban service systems. Section 2 describes the structure of the system while Section 3 contains a brief description of techniques for demand forecasting. An example of planning in the solid waste subsystem is given in Section 4. System integration is described in Section 5. Discussion and conclusions are contained in Section 6.

2.0 STRUCTURE OF THE SYSTEM

The organization pattern of the urban service system can be represented by a two-level hierarchy. The administrative structure of the city or local government constitutes the primary level in the organization. Each of the public utility systems (e.g., water distribution, solid waste, etc.) is defined to be a subsystem in the secondary level of the organization pattern. The two levels together are defined to be the urban service system [Fig. 1].

Funding for major items of the subsystem plans is decided upon by the primary level based on estimates put forward by the subsystems. There is little or no exchange of information between the subsystems. Thus the objective of the primary level is to allocate the resources taking into account the interactions and trade-offs between individual subsystems. The basis for resource allocation requires evaluation of several quantifiable and non-quantifiable criteria. An example of a quantifiable criterion is the cost of a plan while a non-quantifiable criterion is the level of service or the quality of life resulting from a plan or a set of plans.

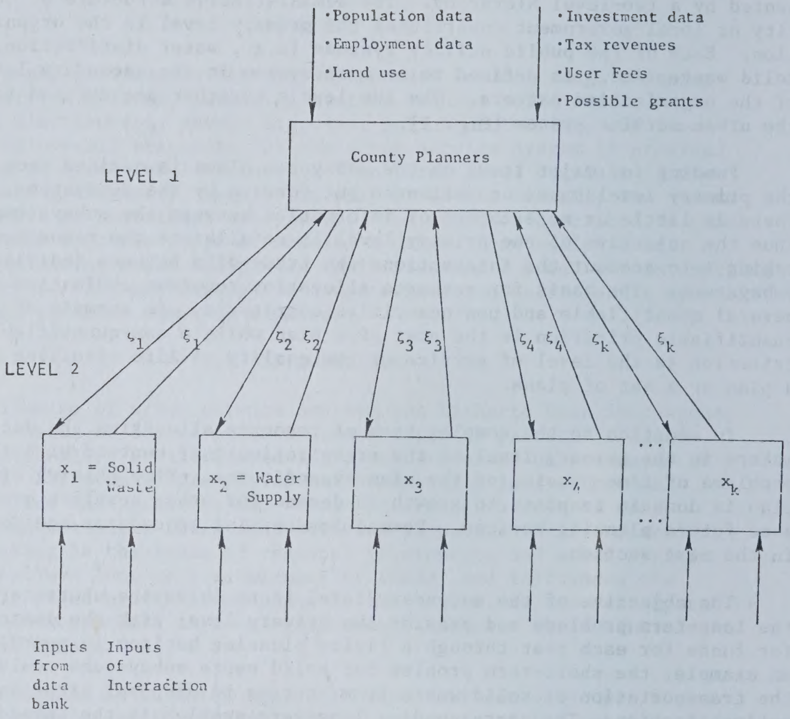
In addition to the complex task of resource allocation the decision makers in the primary level of the organization must contend with the problems of time phasing of the plan expenditures. Time phasing of the plan is done in response to growth in demand for urban services over some future planning horizon. Demand forecasting procedures are described in the next section.

The objective of the secondary level is to solve the short-term and the long-term problems and provide the primary level with the demands for funds for each year through a finite planning horizon (n years). As an example, the short-term problem for solid waste subsystems could be the transportation of solid waste from sources to disposal sites in an optimal fashion. The corresponding long-term problem is the imbedding of the transportation problem in a dynamic program over the next n years to determine the facility expansion program and associated costs.

3.0 DEMAND FORECASTING

The main emphasis in forecasting the demand for urban services is placed on obtaining accurate estimates of population and employment for each disaggregated land parcel. Once these figures are known, conversion factors regarding per capita requirements are fairly well documented for various land use categories. In the case of industrial demand, equivalent parameters based upon employment levels and standard industrial codes are available. Thus, for example, the amount of waste generated per employee by the pulp and paper industry is a known parameter.

Population is accurately tabulated by census tract every ten years and total employment figures are published monthly by state agencies for each county within its jurisdiction. The main problem is to determine population on a yearly basis so that population movements and growth trends are detected, and to distribute the total monthly employment data over the industries and disaggregated land units which are considered. Depending on the size of the region considered and the sensitivity desired,


·Population data
·Employment data
·Land use
·Investment data
·Tax revenues
·User Fees
·Possible grants
County Planners
LEVEL 1
LEVEL 2
ζ_1 ξ_1 ζ_2 ξ_2 ζ_3 ξ_3 ζ_4 ξ_4 ζ_k ξ_k
x_1 = Solid Waste
x_2 = Water Supply
x_3
x_4
x_k
...
Inputs from data bank
Inputs of Interaction

ζ_i = Information re:allocation of funds, user fee etc.

ξ_i = Information about supply, and demand for funds.

FIGURE 1

4.0 SUBSYSTEM PLANNING

As an example of facility planning consider the solid waste subsystem. The objective of solid waste management planning is to determine an effective regional solid waste expansion plan from a set of available alternatives over a planning period. The facilities to be added fall into two categories: processing facilities and disposal facilities. Examples of processing facilities are: (i) Transfer Station, (ii) Incinerator and an example of a disposal facility is a sanitary landfill. For household and commercial wastes the study area is disaggregated into planning analysis zones (PAZ). Each of these PAZs is considered as a waste source. The residential and commercial waste generation forecast for each zone is computed as a function of the land use, population, and employment data. The modeling assumes that all the waste generated at a source is transported by vehicle to either a solid waste processing or disposal facility. In the case of household and commercial waste, it is assumed that all the waste is hauled from the population centroid of the PAZ. The computer models the entire waste flow situation in any year by determining the least cost path for the yearly waste flow from each source to final disposal [Fig. 2].

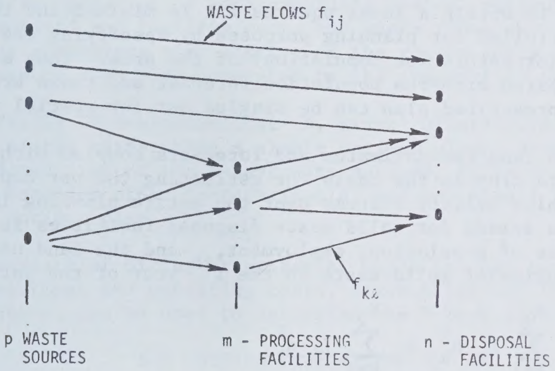


FIGURE 2

In general, the planning procedure operates in the following way: It starts in the first year of the planning period with the existing system, i.e., processing and disposal facilities, and simulates the operation of the system year by year, computing yearly transportation and facility operation costs and waste flows. It develops alternative expansion plans by adding new disposal facilities whenever more disposal capacity is needed or can be economically added to the system. Finally, it calculates the cost of each alternative time-phased plan and determines the least costly plan that satisfies demand.

Imbedded in the solution to the time-phased facility expansion problem is the annual system operation problem, which is solved to determine waste flows and costs of the solid waste facilities available during any year of the planning period. The problem that is solved is a special type of transshipment problem. The objective of solving the problem is to "model" the actual waste flow situation where the waste flow from each source will take the path of least cost to final disposal.

reasonable levels of disaggregation for land use intensity are the section, quarter section, sixteenth section, and census tract or block group (a section consists of 640 acres). One means of obtaining this forecasting information is by employing an extensive land use model. Many such models have been developed and adapted to various regions.

The object of making land use projections using analytical models is to allocate urban economic and industrial activity to regions within the planning area at different points in time over the planning horizon. These models utilize many assumptions about housing demands, consumer shopping and travel characteristics, industrial locational decisions, investment, transportation and land use policies. The idea is to assume varying rates of economic growth and to make projections for the region using the land use model. These regional projections are allocated to the various subregions where the size of any subregion is related to the level of disaggregation desired.

If land use models are not available for activity allocation, it is possible to use a pragmatic approach. It has been determined from data analysis that the growth in most disaggregated land areas (for example, a census tract) can be fairly accurately reproduced by a Gompertz curve. The data required to obtain a least squares fit is minimal and the growth can be controlled for planning purposes by specifying the ultimate holding capacity (or saturation population) of the area. Thus actual growth can be compared with the population forecast and those areas differing from a prescribed plan can be singled out for special investigation.

Disaggregated land use estimates and forecasts coupled with demographic and employment data provide the basis for estimating the per capita demands in the public utility systems over the entire planning interval. As an example, the demand for solid waste disposal facilities is considered. Given the forecasts of population, employment, and the land use in a region, the generation of solid waste in the i^{th} year of the period can be expressed as

$$S_j^i = R\, a_j^i + \sum_{k=1}^{K_i} C_k\, b_{kj}^i + \sum_{k=1}^{K_2} I_k\, d_{kj}^i \qquad (2)$$

where R, C and I are the waste generation multipliers for residential, commercial and industrial sources and

S_j^i = solid waste generated in the i^{th} year in the j^{th} section of the planning region;

a_j^i = population in year i in region j;

b_{kj}^i = equivalent population of k^{th} type commercial establishment in year i in region j;

d_{kj}^i = total capita employed in the k^{th} industry in j^{th} section in year i;

K_1, K_2 = set of commercial and industrial categories respectively.

The multipliers are obtained from regional data averaged over seasonal cycles. Then S_j^i will yield a forecast of solid waste generated in region j for year i once a, b and d are estimated from the forecasts of employment, population and land use for the corresponding region and year.

Using solutions of the transshipment problem, time-phased expansion plan evaluation is done for a group of input candidate facilities, determining which facilities to use and when to open them so that overall costs for the 20-year planning period are minimized. A method based upon forward dynamic programming is employed to determine a family of "best" solutions.

The crucial aspect in solving the dynamic problem is to cut down on the number of alternative state transitions in propagating a plan from year to year (i.e., at every stage). In order to cut down on the number of alternative state transitions at a stage and hence solutions of the transshipment problem, the underlying assumption in the approach is that no system configuration (state) is changed unless a disposal facility runs out. Hence, a particular configuration is merely propagated forward from year to year by computing overall cost and disposal site usage until "runout" occurs on at least one disposal facility. Then a minimum cost replacement is searched for and propagating continues again.

The methodology used in the development of the solid waste plan offers several important advantages:

a) Yearly operation of each candidate system configuration is simulated and facility costs and waste flows are computed, so that a complete time history of the yearly costs and facility usage of each plan generated by the computer is obtained.

b) A family of N-minimum cost (N, given) alternative plans is generated rather than a single optimum plan. Having this ordered list of the N-best alternatives, it is then possible to consider other factors (social and political) in choosing an optimum plan.

c) A variety of different criteria, which include combinations of investment and operating costs, discounting and inflation factors, can be used to determine the N-best plans.

5.0 SYSTEM INTEGRATION

In a general sense, most urban planning projects have an overall goal of providing a maximum level of service to the public at a given cost, or alternatively of providing a given level of service at a minimum cost. In either of these cases, it is usually a straightforward procedure to determine the system cost once a candidate system design has been obtained. The most difficult task is to interpret analytically the meaning of the term "level of service".

In order to be more specific, consider the typical case of a regional government faced with the problem of budget allocation among several public facilities and wishing to distribute its funds to maximize the level of service. This problem can be written as

$$\max_{y_{ik}} z_k = \sum_{i=1}^{n} \Psi_{ik} (y_{ik})$$

subject to

$$\sum_{i=1}^{n} \underline{y}_{ik} \leq \sum_{i=1}^{n} y_{ik} \leq \sum_{i=1}^{n} \overline{y}_{ik}$$

$$\sum_{i=1}^{n} y_{ik} = x_k$$

where

x_k = the total amount of funds to be allocated in the k^{th} year

y_{ik} = the amount of funds allocated to the i^{th} subsystem in the kth year

$\Psi_{ik}(\cdot)$= the cost-effectiveness function for the i^{th} subsystem in the k^{th} year

$\overline{y}_{ik}$ = maximum requested allocation of i^{th} subsystem in k^{th} year

$\underline{y}_{ik}$ = minimum requested allocation of i^{th} subsystem in k^{th} year

This formulation implies that each public utility subsystem determines an optimized expansion plan over a multi-year horizon. This planning optimization yields a need for funds during each year of the planning horizon (say the k^{th}, $0 \leq k \leq T$) where this demand for funds is determined independently of the requirements of any other public utility subsystems. Clearly this problem must be solved over a multiperiod planning horizon in order to determine the optimal time phasing of investment to meet forecasted demand requirements. The decision on when to actually expend these funds, however, is made on a year by year basis and must include trade-offs between competing needs for the available funds, political factors, etc. Thus the budget allocation problem posed above arises yearly and may be solved for several values of x_i corresponding to various policies; for example no new taxes, a small tax increase, a moderate tax increase, etc. The bounds $\underline{y}_{ik}$ and $\overline{y}_{ik}$ correspond respectively to the funding requirements for policies of no expansion for the i^{th} subsystem in year k and full expansion for the i^{th} subsystem in year k.

The cost effectiveness function Ψ_{ik} $(\cdot)$ is determined by fitting a continuous curve such as an S-shaped or negative exponential function to discrete values of cost-effectiveness relations corresponding to alternative expansion plans for each subsystem. For details of the methods to determine $\Psi_{ik}(\cdot)$ please refer to [1].

For the case where the budget allocation is a continuous variable, the solution of the static allocation problem is to allocate an amount of funds y^*_{ik} so that the marginal effectiveness is the same for all subsystems.

6.0 DISCUSSION

The methodology described in this paper was developed and applied to long range planning of urban service systems in Snohomish County, Washington. An efficient data base was designed using the County Assessor's Records, census and employment data, and operational data for the subsystems. Long range plans were developed using procedures similar to those summarized in Section 4. Four subsystems were considered: solid waste, water supply, sewage, and storm drainage. Currently, these plans are being considered for implementation.

System integration met with limited success, not because the methodology was not implementable, but because the political interactions between city and county governments was more complex than anticipated. This aspect of the problem would benefit from further research in decision making in large scale systems. For details of the planning procedures, reference is made to [2].

REFERENCES

1. Rao, H.S. "Utility Theory, Cost Effectiveness and Systems Integration in Urban Planning", TN-ORD-71-8, Systems Control, Inc., Palo Alto, California 94304.
2. WASH USE-1 Final Report, Volumes I-VII, Submitted by Systems Control, Inc., Palo Alto, California to Snohomish County, Washington, June 1973.

COMPUTATIONAL METHODS FOR THE STEADY-STATE KALMAN FILTER IN HIGH ORDER NAVIGATION MODELS*

DAVID L. KLEINMAN**
University of Connecticut
Storrs, Connecticut

ALLAN J. KLEINMAN
Dynamics Research Corp.
Wilmington, Massachusetts

ABSTRACT

Current large scale models for inertial navigation systems include generally 25-40 states for describing gyroscope dynamics, external error sources, measurement characteristics, etc. In such applications the computational burdens associated with obtaining the optimal steady-state Kalman filter and covariance matrix often become excessive. The problem is aggravated by the existance of various combinations of unobservable, "perfectly" observable, and marginally stable states. In addition the measurement time intervals are often not constant, but vary periodically in length.

This paper describes the various computational algorithms that were developed for application to Kalman filtering in such large scale navigational systems. Our numerical experience with the algorithms is discussed.

INTRODUCTION

Current linearized representations for accurately modeling inertial navigation systems such as SINS (Ships Inertial Navigation System) or ESGN (Electrostatically Supported Gyro Navigator) contain generally 25-40 states that describe gyroscope and accelerometer dynamics, external error sources, measurement characteristics, etc. Because of high accuracy needs in the mission requirements of such systems, it is imperative that near optimal use be made of external measurements to update, on-line, estimates of vehicle position, velocity, etc. In such applications, a Kalman filter for optimally processing incoming (discrete) measurements is a natural suggestion. However, due to on-board computer limitations, such an approach is unfeasible and suboptimal filtering schemes are necessary.

To assess the performance of different suboptimal filters in the predesign phase, a yardstick is needed. Thus, an optimal Kalman filter is used to determine the minimum theoretically attainable error covariance. Suboptimal reduced state filters are then designed to achieve accuracies within 10-20% of optimal. However, solving the optimal steady-state filtering equations off-line is a non-trivial matter for such systems. Numerous states, representing inertial error sources, have purely sinusoidal components at earth rate frequency. Depending on vehicle latitude, some of

---

*This work was supported by the Dept. of the Navy, Special Projects Office SP-24.

**Consultant, Dynamics Research Corporation.

these states may be unobservable, or may be perfectly observable (e.g. constant bias) in the steady-state. External measurements (i.e. "fixes") are usually made periodically with measurement interval T. However, in many cases rate information is lost if T is too large, and estimation accuracy degrades. In lieu of reducing T an alternate measurement sequence is often employed with measurement time intervals T_1, T_2, T_1, T_2,... where $T_2 \simeq T$ and $T_1 << T_2$. Obtaining the steady-state (periodic) solution of the variance equation via forward time integration is time-consuming and alternative methods are needed.

This paper describes the various computational algorithms that were used to solve the optimal filtering problems in such large scale navigational systems. Our experiences with the algorithms is discussed.

PROBLEM DESCRIPTION

A linearized model for describing the continuous time evolution of an inertial navigation system may be written in the form

$$\underline{\dot{x}}(t) = \underline{Fx}(t) + \underline{w}(t) \qquad 0 \leq t < \infty \tag{1}$$

where $\underline{x} = \text{col}(x_1, x_2, \ldots x_n)$ are the various states of interest. Common practice is to treat the first n_1 states as gyroscope dynamical states and the last n_2 states as noise shaping states ($n = n_1 + n_2$). Thus the structure of $\underline{F}$ is typically

$$\underline{F} = \left[\begin{array}{c|c} \underline{F}_{11}(n_1 X n_1) & \underline{F}_{12}(n_1 X n_2) \\ \hline \underline{0} & \underline{F}_{22}(n_2 X n_2) \end{array}\right] \tag{2}$$

The submatrix $\underline{F}_{22}$ is composed, in part, of numerous 2X2 sub-blocks of the form

$$\underline{F}_i = \left[\begin{array}{c|c} 0 & -\omega_e \\ \hline \omega_e & 0 \end{array}\right] \tag{3}$$

where ω_e = earth rotation frequency. In Eq. (1), $\underline{w}(t)$ is a zero mean white Gaussian noise with

$$E\{\underline{w}(t)\ \underline{w}'(\tau)\} = \underline{W}\delta(t-\tau) \tag{4}$$

where $\underline{W}$ is a positive semi-definite (diagonal) matrix. In most applications $n_1 \simeq$ 6-8 while $n_2 \simeq 20$, i.e. the analyst attempts to model most of the error sources that effect the gyroscope motion and accuracy.

External measurements of vehicle position (e.g. latitude and longitude) and, possibly, heading or velocity are made at discrete times t_1, t_2,

t_3,.. with the m-measurements given by

$$\underline{y}(t_i) = \underline{Hx}(t_i) + \underline{v}(t_i) \qquad i=1,2,... \tag{5}$$

where $\underline{v}$ is a white Gaussian measurement noise with covariance

$$E\{\underline{v}(t_i)\ \underline{v}'(t_j)\} = \underline{R}\delta(t_i - t_j) \tag{6}$$

The objective is to utilize the measurements $\underline{y}(t_i)$ to obtain an estimate $\hat{\underline{x}}(t)$ of $\underline{x}(t)$ with minimum estimation error, $\underline{e}(t) = \underline{x}(t) - \hat{\underline{x}}(t)$. Knowledge of the minimum estimation error can then serve as a yardstick for evaluating the performance of various suboptimal estimation schemes. Usually we are interested in performance in the steady-state, i.e. after initial transients have died out.

THE OPTIMAL FILTER, $t_{i+1} - t_i = T$

For the case when the measurement time interval is constant, i.e. $t_{i+1} - t_i = T$, the optimal estimate is generated recursively by a Kalman filter [1]

$$\hat{\underline{x}}_i^{+} = \hat{\underline{x}}_i^{-} - \underline{K}_i(\underline{H}\hat{\underline{x}}_i^{-} - y_i) \quad ; \quad \hat{\underline{x}}_0^{-} = \underline{0} \tag{7}$$

$$\hat{\underline{x}}_{i+1}^{-} = \underline{\Phi}\hat{\underline{x}}_i^{+} \tag{8}$$

where $\hat{\underline{x}}_i^{-}$ is the best estimate of $\underline{x}(t_i)$ prior to obtaining the measurement $\underline{y}_i = \underline{y}(t_i)$, and $\hat{\underline{x}}_i^{+}$ is the updated estimate at t_i. The matrix $\underline{\Phi}$ is the open-loop transition matrix

$$\underline{\Phi} = e^{\underline{F}T} \tag{9}$$

Between measurements, the optimal estimate is obtained from

$$\hat{\underline{x}}(t + t_i) = e^{\underline{F}t}\ \hat{\underline{x}}_i^{+} \qquad 0 \leq t < T \tag{10}$$

The gain matrix $\underline{K}_i$ is obtained from

$$\underline{K}_i = \underline{M}_i\underline{H}'(\underline{R} + \underline{H}\underline{M}_i\underline{H}')^{-1} = \text{"gain" at time } t_i \tag{11}$$

where, recursively

$$\underline{P}_i = (\underline{I} - \underline{K}_i\underline{H})\underline{M}_i(\underline{I} - \underline{K}_i\underline{H})' + \underline{K}_i\underline{R}\underline{K}_i' \tag{12}$$

$$\underline{M}_{i+1} = \underline{\Phi}\ \underline{P}_i\underline{\Phi}' + \underline{Q} \tag{13}$$

for i=0,1,... and $\underline{M}_0$ is a given matrix. The matrix $\underline{Q}$ is

$$\underline{Q} = \int_0^T e^{\underline{F}t}\ \underline{W}e^{\underline{F}'t}\ dt \tag{14}$$

The matrices $\underline{M}_i$ and $\underline{P}_i$ are the estimation error covariance matrices with

$$E\{(\hat{\underline{x}}_i^- - \underline{x}_i)(\hat{\underline{x}}_i^- - \underline{x}_i)'\} = \underline{M}_i \tag{15a}$$

$$E\{(\hat{\underline{x}}_i^+ - \underline{x}_i)(\hat{\underline{x}}_i^+ - \underline{x}_i)'\} = \underline{P}_i \tag{15b}$$

Thus, $\underline{M}_i$ is the error covariance prior to the fix at time t_i, and $\underline{P}_i$ is the error covariance after the fix, so $\underline{P}_i < \underline{M}_i$.

In the steady-state, assuming stabilizability of $(\underline{A}',\underline{H}')$, $\underline{P}_i \to \underline{P}$, $\underline{M}_i \to \underline{M}$ and $\underline{K}_i \to \underline{K}$. Thus, the optimal steady-state filtering problem requires the solution to the well-known nonlinear (algebraic) equation, i.e. the variance equation

$$\underline{M} = \underline{\Phi}(\underline{I} - \underline{K}\underline{H})\underline{M}(\underline{I} - \underline{K}\underline{H})'\underline{\Phi}' + \underline{\Phi}\underline{K}\underline{R}\underline{K}'\underline{\Phi}' + \underline{Q} \tag{16}$$

$$\underline{K} = \underline{M}\underline{H}'(\underline{R} + \underline{H}\underline{M}\underline{H}')^{-1} \tag{17}$$

An alternate formula for $\underline{M}$ can be obtained as

$$\underline{M} = \underline{\Phi}[\underline{M} - \underline{M}\underline{H}'(\underline{R} + \underline{H}\underline{M}\underline{H}')^{-1}\,\underline{H}\underline{M}]\,\underline{\Phi}' + \underline{Q} \tag{18}$$

NUMERICAL SOLUTION OF VARIANCE EQUATION

There are several methods in the literature for solving Eq. (18). The most straightforward way is to integrate Eqs. (12)-(13) forward in time until $\underline{M}_i \to \underline{M}$. However, this is usually a slowly (linearly) convergent process, and when faced with a large number of states it becomes unattractive, computationally. Another method for solving (18) is based on an eigenvalue-eigenvector scheme [2]. However, it is the eigenvectors of a 2nX2n matrix corresponding to the Hamiltonian equations that must be found. This fact, coupled with the existance of repeated roots and unobservable modes, lead us to bypass such a scheme for finding $\underline{M}$.

An iterative method for obtaining $\underline{M}$ was selected [3]. The algorithm is given by

$$\underline{M} = \lim_{k\to\infty} \underline{V}_k \tag{19}$$

where $\underline{V}_k$, k=1,2,... is the solution of the linear equation

$$\underline{V}_k = \underline{\Phi}(\underline{I} - \underline{K}_k\underline{H})\underline{V}_k(\underline{I} - \underline{K}_k\underline{H})'\underline{\Phi}' + \underline{\Phi}\underline{K}_k\underline{R}\underline{K}_k'\underline{\Phi}' + \underline{Q} \tag{20}$$

with $$\underline{K}_k = \underline{V}_{k-1}\underline{H}'(\underline{R} + \underline{H}\underline{V}_{k-1}\underline{H}')^{-1} \tag{21}$$

This is a convergent algorithm provided $\underline{V}_0$ (or $\underline{K}_1$) is chosen such that $\underline{\Phi}(\underline{I} - \underline{K}_1\underline{H})$ has eigenvalues within the unit circle. A useful choice for $\underline{V}_0$, even in cases when $\underline{\Phi}$ is stable, is given by [4]

$$\underline{V}_0 = \underline{\Phi}^N \left[\sum_{i=0}^{N-1} (\underline{\Phi}')^i\underline{H}'\underline{R}\underline{H}\underline{\Phi}^i\right]^{\#} (\underline{\Phi}')^N \tag{22}$$

where $N \geq n$ and where # denotes the Penrose psuedo-inverse.

Each iteration (20)-(21) involves the solution of a linear equation of the form

$$\underline{X} = \underline{\bar{\Phi} X \bar{\Phi}'} + \underline{\bar{Q}} \qquad (23)$$

where $|\lambda(\underline{\bar{\Phi}})| \leq 1$. While there are numerous methods for solving (23), it was necessary to use a scheme that would work even for cases when some $|\lambda_i(\underline{\bar{\Phi}})| \equiv 1$. Thus $\underline{X}$ was obtained as the summation,

$$\underline{X} = \sum_{i=0}^{\infty} \underline{\bar{\Phi}}^{i}\ \underline{\bar{Q}}(\underline{\bar{\Phi}}')^{i} \qquad (24)$$

where this sum was evaluated iteratively according to

$$X_0 = \underline{\bar{Q}} \ , \quad \underline{P} = \underline{\bar{\Phi}} \ , \quad k=0 \quad \text{(initialization)} \qquad (25a)$$

$$\underline{X_{k+1}} = \underline{X_k} + \underline{PX_kP'} \qquad (25b)$$

$$\underline{P} = \underline{P}^2 \qquad (25c)$$

$$k = k + 1$$

The process is continued until $||\underline{X_{k+1}} - \underline{X_k}|| \simeq 0$. Since k iterations has summed 2^{k-1} terms, convergence is rapid.

In all of the inertial navigation filtering problems considered to date, ranging from 23-45 states, the doubly-iterative scheme outlined above has always worked reliably. The following observations have been made converning the algorithm

1. Convergence, which becomes quadratic as $\underline{V_k} \rightarrow M$, generally occurs within 10 iterations.
2. Convergence occurs despite the existance of unobservable modes, even if they are marginally stable.
3. The algorithm cannot converge to an indefinite solution of (18).
4. In cases where some states can be estimated perfectly (i.e. $\underline{w}(t)$ does not affect all modes) the algorithm converges to the proper positive semi-definite solution of (18).
5. The computation time increases as n^3.
6. Round-off errors, sometimes troublesome when solving the continuous time variance equation, presented no problem in the present discrete case.

COMPUTING THE MATRICES $\underline{\Phi}$ AND $\underline{Q}$

In the process of discretizing the continuous time system (1) it is necessary to compute the matrices $\underline{\Phi}$ and $\underline{Q}$. Normally, this is a straightforward process via series summation or other schemes. However, when n is large, considerable computer time can be wasted by inefficient algorithms. The methods that we selected for evaluating $\underline{\Phi}$ and $\underline{Q}$ are described below.

1. Computing Φ

A useful method for evaluating the matrix $e^{\underline{F}T}$ is based on the n-th order (scalar) Padé Approximation to e^x

$$e^x \doteq f_n(x) = \frac{N_n(x)}{D_n(x)} \tag{26}$$

where

$$N_n(x) = 1 + \sum_{i=1}^{n} \frac{n(n-1)\,\ldots\,(n-i+1)}{2n(2n-1)\ldots(2n-i+1)} \cdot \frac{x^i}{i!} \tag{27}$$

and $D_n(x) = N_n(-x)$. The Padé approximation is highly accurate with

$$\left|e^x - f_n(x)\right| = \frac{o(x^{2n+1})}{(2n+1)!} \tag{28}$$

The matrix exponential $e^{\underline{F}\delta}$ is approximated as $e^{\underline{F}\delta} = f_n(\underline{F}\delta)$ where the time step $\delta=T/k$ is chosen such that $||\underline{F}\delta||$ is commensurate with desired accuracy. $e^{\underline{F}T}$ is then obtained by successive multiplications of $e^{\underline{F}\delta}$. The above method with n=2 has been suggested by Davison [5].

The accuracy of the Padé scheme is tempered by the requirement for several temporary storage matrices and the need for a matrix inversion. Thus, an alternate approximation, based on the n-th order Chebyshev expansion of e^{2x} was used [6-7], where

$$e^{2x} \doteq g_n(x) = I_0(2) + 2\sum_{m=1}^{n} I_m(2)\, T_m(x) \qquad |x| \le 1 \tag{29}$$

In Eq. (29), I_m is the modified Bessel function of the first kind,

$$I_m(2) = \sum_{k=0}^{\infty} \frac{1}{k!\,(m+k)!} \tag{30}$$

and

$$T_m(x) = \frac{m}{2} \sum_{s=0}^{2s\le m} \frac{(-1)^s}{m-s} \binom{m-s}{s} (2x)^{m-2s} \tag{31}$$

The error in the approximation (29) is uniform in x, with

$$\left|e^{2x} - g_n(x)\right| \le 2I_n(2)\,\frac{n+1}{n} = o\left(\frac{1}{n!}\right) \tag{32}$$

In our work we chose n=12 and expanded Eq. (29) as

$$e^{2x} \approx 1 + \sum_{i=1}^{n} c_i x^i \tag{33}$$

Where the coefficients c_i are given in Table 1. To obtain the matrix $e^{\underline{F}T}$, $M \geq 1$ is found such that $||\underline{F}T/2^M|| \leq 1$, and then

$$\underline{X} = e^{2 \cdot \frac{\underline{F}T}{2^M}} = e^{\underline{F}T/2^{M-1}} = \sum_{i=0}^{n} c_i \left(\frac{\underline{F}T}{2^M}\right)^i \tag{34}$$

is computed. Finally, $e^{\underline{F}T}$ is obtained by successive squaring $\underline{X}^{2^{M-1}}$.

TABLE 1: COEFFICIENTS IN CHEBYSHEV EXPANSION OF e^{2x}

c_1 = 1.999999997E 0
c_2 = 1.999999998E 0
c_3 = 1.333333458E 0
c_4 = 6.666667048E-1

c_5 = 2.666656721E-1
c_6 = 8.888866040E-2
c_7 = 2.540022245E-2
c_8 = 6.349856948E-3

c_{09} = 1.405308746E-3
c_{10} = 2.812383398E-4
c_{11} = 5.575070384E-5
c_{12} = 9.232920486E-6

2. Computing $\underline{Q}$

The noise integral (14) was computed according to

$$\underline{Q} = \sum_{i=0}^{2^k-1} (e^{\underline{F}\delta})^i \int_0^\delta e^{\underline{F}\sigma} \underline{w} e^{\underline{F}'\sigma} d\sigma \, (e^{\underline{F}'\delta})^i \tag{35}$$

where $k>0$ is chosen such that $||\underline{F}T/2^k|| \leq 0.5$. $\underline{X} = e^{\underline{F}\delta}$ is computed by (34) and the integral term in Eq. (35) is obtained by summing the first 16 terms in the series

$$\underline{Q}_0 = \underline{W}\delta + (\underline{F}\underline{W} + \underline{W}\underline{F}')\frac{\delta^2}{2} + (\underline{F}^2\underline{W} + 2\underline{F}\underline{W}\underline{F}' + \underline{W}\underline{F}'^2)\frac{\delta^3}{3!} + \ldots \tag{36}$$

This series is easy to add as each term $\underline{Y}_k$ follows from the previous one

$$\underline{Y}_k = \frac{\delta}{k}(\underline{F}\underline{Y}_{k-1} + \underline{Y}_{k-1}\underline{F}') \quad ; \quad \underline{Y}_0 = \delta\underline{W} \tag{37}$$

the summation (35) is obtained by letting $\underline{Q} = \underline{Q}_k$ where

$$\underline{Q}_i = \underline{X}^{2^{i-1}} \underline{Q}_{i-1} \underline{X}'^{2^{i-1}} + \underline{Q}_{i-1} \quad ; \quad i=1,2,\ldots k \tag{38}$$

and $\underline{Q}_0$ is given by Eq. (36).

NON-CONSTANT MEASUREMENT INTERVALS

In certain applications the time interval between observations $\underline{y}_{i+1}$ and $\underline{y}_i$ is not constant, yet the measurement sequences have a regular pattern. For example, a measurement is made at $t_0=0$, another at $t_1=T_1$, another at $t_2=T_1+T_2$, another at $t_3=T_1+T_2+T_1$, etc. Thus, the measurement intervals vary as $T_1, T_2, T_1, T_2, \ldots$ were we assume $T_1<T_2$.

In the steady-state as $t \to \infty$, the solution of the optimal filtering equations (11-13) is periodic with period $T=T_1+T_2$, i.e. $\underline{M}(t_{i+3}) = \underline{M}(t_i)$ and $\underline{K}(t_{i+3}) = \underline{K}(t_i)$. Referring to Fig. 1, which portrays "steady-state" behavior of the error covariance, we seek the matrices $\underline{M}_1$, $\underline{P}_1$, $\underline{M}_2$, $\underline{P}_2$. Note that only $\underline{M}_2$, is needed since the others follow from Eqs. (12)-(13).

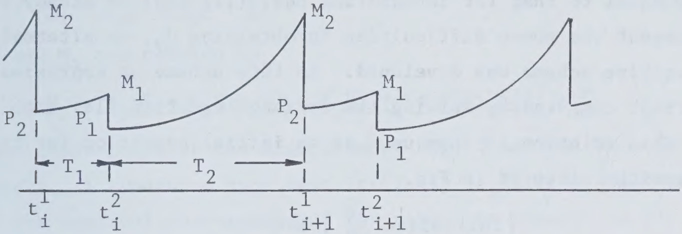


Fig. 1. Steady-state covariance relationships

One method to obtain $\underline{M}_2$ is by integrating Eqs (12-13) to steady-state, i.e. until $\underline{M}_2(t_{i+3}) \simeq \underline{M}_2(t_i)$. However, this is inefficient from a computational viewpoint. Another method is to formulate a 2n-dimensional filtering problem by defining an augmented state $\tilde{\underline{x}}_i = \text{col}\,[\underline{x}(t_i^1), \underline{x}(t_i^2)]$. To obtain a reccurence relation between $\tilde{\underline{x}}_i$ and $\tilde{\underline{x}}_{i+1}$ we note that

$$\underline{x}(t_i^2) = \underline{\Phi}_1\, \underline{x}(t_i^1) + \underline{w}_1 \tag{39}$$

$$\underline{x}(t_{i+1}^2) = \underline{\Phi}_1\, \underline{x}(t_{i+1}^1) + \underline{w}_1 \tag{40}$$

$$\underline{x}(t_{i+1}^1) = \underline{\Phi}_2\, \underline{x}(t_i^2) + \underline{w}_2 \tag{41}$$

where each $\underline{w}_i$ is an (independent) white-noise sequence, with covariance $\underline{Q}_i$,

$$Q_i = \int_0^{T_i} e^{\underline{F}\sigma}\, \underline{W} e^{\underline{F}'\sigma}\, d\sigma \quad , \quad \underline{\Phi}_i = e^{\underline{F}T_i} \quad ; \; i=1,2 \tag{42}$$

Thus, from Eqs. (39)-(41) we obtain

$$\tilde{\underline{x}}_{i+1} = \tilde{\underline{\Phi}}\tilde{\underline{x}}_i + \tilde{\underline{w}} \tag{43}$$

where $\tilde{\underline{w}}$ has covariance $\tilde{\underline{Q}}$ with

$$\tilde{\underline{\Phi}} = \left[\begin{array}{c|c} \underline{0} & \underline{\Phi}_2 \\ \hline \underline{0} & \underline{\Phi}_1\underline{\Phi}_2 \end{array}\right] \quad ; \quad \tilde{\underline{Q}} = \left[\begin{array}{c|c} \underline{Q}_2 & \underline{Q}_2\underline{\Phi}_1' \\ \hline \underline{\Phi}_1\underline{Q}_2 & \underline{Q}_1 + \underline{\Phi}_1\underline{Q}_2\underline{\Phi}_1' \end{array}\right] \tag{44}$$

Instead of considering single measurements at times t_i^1, t_i^2, etc. we consider a double measurement $\underline{y} = \text{col}\,[y(t_i^1), y(t_i^2)]$ made at time t_i^2, thus

$$\tilde{\underline{y}}_i = \tilde{\underline{H}}\tilde{\underline{x}}_i + \tilde{\underline{v}}_i = \left[\begin{array}{c|c} \underline{H} & \underline{0} \\ \hline \underline{0} & \underline{H} \end{array}\right] \tilde{\underline{x}}_i + \left[\begin{array}{c} \underline{v}_1 \\ \hline \underline{v}_2 \end{array}\right] \tag{45}$$

with cov $[\tilde{\underline{v}}_i] = \text{diag}\,[\underline{R},\underline{R}]$.

If we now consider the steady-state variance equation (18) with the replacements $\underline{\Phi} \rightarrow \tilde{\underline{\Phi}}$, $\underline{Q} \rightarrow \tilde{\underline{Q}}$, etc., the resulting 2nX2n matrix $\tilde{\underline{M}}$ has the property that the upper left nXn sub-block is the requisite matrix $\underline{M}_2$.

The above method for obtaining M_2 was tried for n=23 states. However, the time required to solve the resulting 46X46 variance equations was approximately equal to that for integrating Eqs. (12)-(13) to steady-state.

To circumvent the above difficulties in obtaining M_2, an alternate, two stage iterative scheme was developed. In this scheme an approximation to $M_2 = M_2^0$ is first computed by solving the variance equation (18) with $\Phi = \Phi_2$, $Q = Q_2$. This solution is then used as an initial condition for the iterative algorithm pictured in Fig. 2.

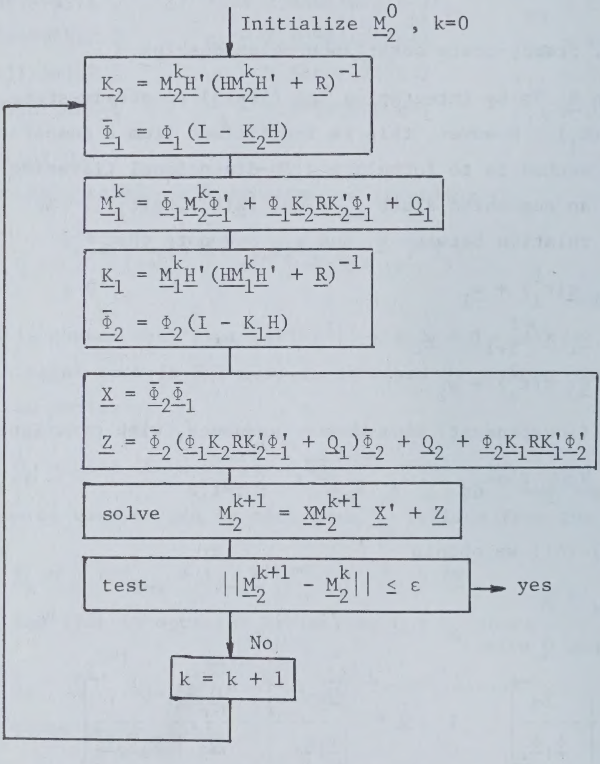

Initialize M_2^0 , k=0
$K_2 = M_2^k H' (HM_2^k H' + R)^{-1}$
$\bar{\Phi}_1 = \Phi_1 (I - K_2 H)$
$M_1^k = \bar{\Phi}_1 M_2^k \bar{\Phi}_1' + \Phi_1 K_2 R K_2' \Phi_1' + Q_1$
$K_1 = M_1^k H' (HM_1^k H' + R)^{-1}$
$\bar{\Phi}_2 = \Phi_2 (I - K_1 H)$
$X = \bar{\Phi}_2 \bar{\Phi}_1$
$Z = \bar{\Phi}_2 (\Phi_1 K_2 R K_2' \Phi_1' + Q_1) \bar{\Phi}_2' + Q_2 + \Phi_2 K_1 R K_1' \Phi_2'$
solve $M_2^{k+1} = X M_2^{k+1} X' + Z$
test $||M_2^{k+1} - M_2^k|| \leq \varepsilon$
yes
No
k = k + 1

Fig. 2. Iterative algorithm for obtaining steady-state M_2

This algorithm is a logical extension of that used to solve Eq. (18) since M_2 in the steady state satisfies the equation

$$M_2 = X M_2 X' + Z \tag{46}$$

where

$$X = \Phi_2 (I - K_1 H) \Phi_1 (I - K_2 H) \tag{47}$$

$$\underline{Z} = \underline{\Phi}_2(\underline{I} - \underline{K}_1\underline{H})[\underline{\Phi}_1\underline{K}_2\underline{RK}_2'\underline{\Phi}_1' + \underline{Q}_1](\underline{I} - \underline{K}_1\underline{H})'\underline{\Phi}_2' + \underline{Q}_2 + \underline{\Phi}_2\underline{K}_1\underline{RK}_1'\underline{\Phi}_2' \quad (48)$$

$$\underline{K}_i = \underline{M}_i\underline{H}'(\underline{HM}_i\underline{H}' + \underline{R})^{-1} \quad (49)$$

and $\underline{M}_1$ and $\underline{M}_2$ are related by

$$\underline{M}_1 = \underline{\Phi}_1(\underline{I} - \underline{K}_2\underline{H})\underline{M}_2(\underline{I} - \underline{K}_2\underline{H})'\underline{\Phi}_1' + \underline{\Phi}_1\underline{K}_2\underline{RK}_2'\underline{\Phi}_1' + \underline{Q}_1 \quad (50)$$

The convergence of the algorithm of Fig. 2 has not been proven rigorously. Although we know that initially $|\lambda_i(\underline{\bar{\Phi}}_1)| \leq 1$ with our choice of $\underline{M}_2^0$, the eventual convergence of the algorithm depends on the condition $|\lambda_i(\underline{\Phi}_2\underline{\Phi}_1)| \leq 1$, and whether successive gain matrices generated by the algorithm retain this stability property. These questions are items for further research. However, our numerical experience with the algorithm is highly encouraging:

1. Using the initial iterate $\underline{M}_2^0$, convergence was always found to occur for quite arbitrary T_1, T_2.
2. The scheme is most efficient when the interval times are ordered such that $T_1 < T_2$.
3. Once $\underline{M}_2^0$ is obtained, approximately 3-4 iterations are needed for the algorithm to converge.
4. Extensions to multiple ($r>2$) measurement intervals are easily accomplished and exhibit similar convergence properties.

TIME PROPAGATION AND SPARSENESS TESTING

Once the covariance matrix $\underline{M}$ (or $\underline{P}$) is found, it is usually of interest to obtain the time history of the error covariance $\underline{\Sigma}(t) = E\{[\underline{x}(t) - \hat{\underline{x}}(t)]\cdot[\underline{x}(t) - \hat{\underline{x}}(t)]'\}$ over one entire period. Obtaining $\underline{\Sigma}(t)$ where $\underline{\Sigma}(0) = \underline{P}$ is a straightforward process,

$$\underline{\Sigma}(t+\delta) = e^{\underline{F}\delta}\,\underline{\Sigma}(t)\,e^{\underline{F}'\delta} + \int_0^{\delta} e^{\underline{F}\sigma}\,\underline{W}\,e^{\underline{F}'\sigma}\,d\sigma \quad (51)$$

Integrating Eq. (51) can be time consuming, especially when δ is small, or if n is large. However, due to the structure (2-3) of $\underline{F}$, the matrix $e^{\underline{F}\delta}$ will also have an upper triangular block form where the lower $n_2 X n_2$ sub-block will have a dominant, main diagonal, banded structure. Thus, $e^{\underline{F}\delta}$ can be quite sparse. (Note that $\underline{\Sigma}$ generally need not be very sparse.) To take advantage of sparseness in performing a matrix multiplication $F\delta$ we can tag a priori the non-zero elements of $e^{\underline{F}\delta}$ (and $\underline{\Sigma}$) and perform only the non-zero scalar multiplications. Even if this approach were entirely possible, it may not be desirable in that it renders the computer programs highly specific. Changing noise models, etc., in any engineering study may thus prove to be an annoying reprogramming task.

The method we selected for matrix multiplication $\underline{AB}$ with $\underline{A}$ relatively sparse, was to write $\underline{X}$ as a dyadic summation

$$\underline{X} = \underline{AB} = \sum_{i=1}^{m} \underline{a}_i \underline{b}_i' \quad ; \quad \underline{A} = nXm, \ \underline{B} = mXp \tag{52}$$

where $\underline{a}_i$ and $\underline{b}_i'$ are the columns of $\underline{A}$ and rows of $\underline{B}$, respectively. Each nXp dyad is evaluated one row at a time, e.g. the j-th row of $\underline{a}_i \underline{b}_i'$ is $a_{ij} \underline{b}_i$. Thus, by testing $a_{ij} = 0$ it is possible to avoid p scalar multiplications. As a result, the total number, N, of computer operations (neglecting the test a_{ij} vs. 0.) to multiply $\underline{AB}$ is reduced from nmp to N = (# of non-zero elements of $\underline{A}$)p.

CONCLUSIONS

In this paper we have described computational algorithms that experience proved most useful when solving optimal linear filtering periods for high-order navigation models. When the measurement interval was constant, an iterative method for solving the steady-state variance equation gave reliable performance in all cases studied. For n=23 to 46 states, convergence was generally attained in 10 iterations, even in cases when some states were unobservable (and marginally stable) and when some states were perfectly observable (variance solution becomes p.s.d.).

If measurements are made according to a non-constant fix interval scheme, an extension of the basic iterative algorithm was found most efficient for computing the steady-state (periodic) covariance matrix. Typically, 4 further iterations were necessary once a steady-state (constant) solution was obtained as an initial condition.

In addition to the algorithms for solving the major variance equation, the techniques we used for obtaining the matrix exponential, covariance integral and sparse matrix product were discussed. Attention to these underlying, often neglected, aspects of the overall problem can result in considerable savings of computer time in a practical application.

REFERENCES

1. Bryson, A.E. and Ho, Y.C., Applied Optimal Control, Blaisdell, 1969.
2. Vaughan, D.R., "A Nonrecursive Algebraic Solution for the Discrete Riccati Equation", IEEE Trans. Autom. Control, Vol. AC-15, Oct. 1970.
3. Hewer, G.A., "An Iterative Technique for the Computation of the Steady-State Gains for the Discrete Optimal Regulator", IEEE Trans. Autom. Control, Vol. AC-16, Aug. 1971.
4. Kleinman, D.L., "Stabilizing a Discrete, Constant, Linear System with Application to Iterative Methods for Solving the Riccati Equation", IEEE Trans. Autom. Control, Vol. AC-19, June 1974.
5. Davison, E.J., "A High Order Crank-Nicholson Technique for Solving Differential Equations", Comput. Journal, Vol. 10, Aug. 1967.
6. Hart, J.F., et. al., Computer Approximations, Chapter 3, John Wiley & Sons, Inc.
7. Handbook of Mathematical Functions, p. 374-76, U.S. Dept. of Commerce, Nat'l. Bureau of Standards Applied Math Series 55, June 1964.

LARGE SYSTEMS MODELING FOR COORDINATING ATMOSPHERIC PROCESSES AND ELECTRIC POWER OPERATION *

JAMES GRUHL
M.I.T. Energy Laboratory
Cambridge, Massachusetts 02139

ABSTRACT

At the present levels of coordination, electric power system operation involves system modeling and optimization procedures on a large scale. The possibilities for introducing environmental considerations in the planning and operation of power systems enlarge the scale of the problem, and require the special adaptation of some of the previously used methods. The control procedures for, and the potential gains from, producing electric power in a way more compatible with the environment are discussed here for the minute-by-minute to the yearly range of time scales. Particular attention is given to the coordination of the atmosphere's time-varying dispersive capabilities and the air pollutant production resulting from power system operation.

INTRODUCTION

Attempting to increase energy supplies while simultaneously increasing environmental quality is a problem which has grown quickly into one of major concern. Large scale systems modeling is currently being used on every facet of this energy-environmental dilemma. As it continues to enlarge in scale and in scope finding answers in this problem area requires making special adaptations of existing solution techniques. In addition to the new environmental concerns, models must now incorporate the finiteness of land and resources, and must be capable of predicting effects of scenarios such as economic slowdowns and shortages of energy, food, and capital.

Of course, one must work either with crude aggregations or with small partitions of this general energy-environmental problem. And even the small partition considered here, the coordination of electric power production and atmospheric processes, is itself typically divided into six different hierarchies of activities, as shown in Figure 1. The most far-sighted of these activities is the effort to develop improved technologies. These activities are either government or industry sponsored, or are regulation induced, and studies of them are generally confined to small-scale cost-benefit analyses.

---

* Portions of this work have been sponsored by the Atomic Energy Commission under contract number AT(11-1)-2428 and by the National Science Foundation— RANN Division under grant number GI-32874.

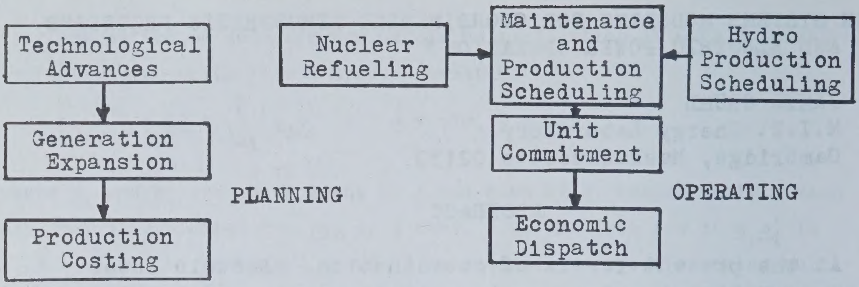


Figure 1 Hierarchy of the activities which can be used to provide better compatibility between electric power production and atmospheric dispersion processes.

In Figure 1 the planning process used by electric utilities to decide on new generation facilities is called generation expansion, and the simulation of these generation expansion schemes is called production costing. It has only been recently (see Ref. 1) that there has been any attempt to coordinate, on a large system scale, the choosing and siting of new facilities with regard to pollution production and dispersion. As is generally the case in large system optimization where the solution space is not tightly constrained (that is, there is a very broad set of feasible solutions) linear programming plays an important role in the solution procedures.

MAINTENANCE AND PRODUCTION SCHEDULING

That short overview of the planning procedures leads us to the large scale systems work being emphasized here. That part of the power system operation problem called the maintenance and production scheduler computes weekly (or monthly) schedules out to a one- to four-year horizon. Some of the major questions addressed by this type of scheduler are:

(1) timing and duration of maintenance outages,
(2) timing and batch sizes of nuclear refuelings,
(3) commitment of interutility power sales and purchases
(4) developing weekly production schedules for the optimal utilization of fixed batches of nuclear, fossil, and hydro energy, and
(5) forecasting, contracting, and leveling of maintenance and refueling manpower.

Ideally, all of these tasks must be performed under the requirements of maintaining system reliability and minimizing expenses (Ref. 2 contains a more detailed accounting of this scheduler).

Historically, crude, manual techniques have been adequate for small power systems, but now whole regions of the country are pooled together and it is becoming obvious that there can be large economic and environmental gains from clever scheduling.

The primary constraint equations in the maintenance scheduler are those which require that the demand for power be met. Thus if

$A_i(k)$ = capacity of i^{th} plant in k^{th} time interval derated to average in the effects of forced outages,

$$u_i(k) = \begin{cases} 1 & \text{if plant } i \text{ is on maintenance at time } k \\ 0 & \text{otherwise,} \end{cases}$$

$M(k)$ = amount of derated generating capacity which may be scheduled out for maintenance at time k,

$I_j(k)$ = interregional power purchase or sale options, measured in megawatts,

$w_j(k)$ = 0 to 1, representing extent of interchange to be used (could be a binary decision variable), then

$$\sum_{\text{all } i} A_i(k)u_i(k) - \sum_{\text{all } j} I_j(k)w_j(k) \leqslant M(k) \qquad (1)$$

Using either a reserve or a risk technique (as described in Refs. 3 and 4), the reliability of a schedule, r, can be incorporated as a variable in the power demand Eq. 1 by substituting for M(k) the expression

$$M(k) = C(k) - D(k) - r \cdot R(k) \qquad (2)$$

where

$C(k)$ = the effective capacity of the system

$D(k)$ = the actual forecast demand (or equivalent demand),

$R(k)$ = the reserve (or equivalent reserve), and

r = the reliability of the schedule.

The linear programming format being used allows for any of the currently used (plus many additional) maintenance scheduling features. For example, if the basic unit of time is two weeks, then maintenance requirements for a given plant i could typically take the form of options such as:

(1) two weeks of maintenance may be started anytime between 44 and 60 weeks from the present, or

(2) four weeks of maintenance may be started anytime within the window from 40 to 56 weeks from the present.

The requirement for some kind of maintenance attention for this plant i then becomes:

$$\sum_{k=22}^{30} u_i^1(k) + \sum_{k=20}^{28} u_i^2(k) = 1 \qquad (3)$$

where u_i^2 represents a maintenance session spanning two time intervals.

Power systems with flexibility in the management of reservoir levels must couple these opportunities with their other production decisions if they wish to maximize performance. If $h_j(k)$ is the extent of power production of the j^th hydro plant at time k, where this quantity is constrained from above by capacity limitations and from below by minimum flow requirements, then

$$WE_j(k-1) + RI_j(k) - h_j(k) - SP_j(k) = WE_j(k) \qquad (4)$$

is the water energy accountage for reservoir j where

$WE_j(k)$ = usable water energy content of the j^th reservoir at time k,

$RI_j(k)$ = inflow into the j^{th} forebay, and

$SP_j(k)$ = intentional or inadvertent spillage.

Both $h_j(k)$ and $SP_j(k)$ are generally introduced into the performance measure equation, and $h_j(k)$ contributes to the power production at time k, wherein it reflects the inefficiency of the water-to-electricity energy conversion.

The number of constraint equations grows very large when equations are included for the possible economic shutdown of some facilities, the coordination of the efforts of different maintenance crews, fixed energy-batch limitations for nuclear, gas and low-sulfur fuels, and a whole set of cycling demand equations to ensure that there will be available adequate amounts of fast responding units.

Performance Measure

A typical objective function equation is used to collect the economic and environmental consequences of the schedules. Among those variables which contribute to the economic cost of schedules are:

(1) operating costs,
(2) interregional power sales
(3) maintenance costs,
(4) refueling costs, and
(5) contract costs and penalties.

The exact form in which these costs are collected is straightforward, and can be found in Ref. 2.

Not as straightforward is the method for weighting the measure of desirability of schedules for environmental reasons. Using environmental weightings, for example, nuclear facilities would tend to be scheduled for refueling in August or September when thermal impacts are severest, and fossil-fueled plants would tend to be maintained at times of the year with high probabilities of stagnating air masses. Such tendencies would help prevent the temporary shutdowns of facilities due to environmental alerts. Some of the meaningful environmental parameters which might be useful involve quantifying:

(1) the amounts of emissions,
(2) the severity of environmental standards which could be met, or
(3) the consumptive use of natural resources.

Perhaps the most difficult, but probably the most useful, environmental measure to establish is the type of measure which reflects the ultimate, real impacts to the surrounding ecosystem (some are described in Refs. 5 and 6). Roughly, these quantifications of impacts take into account the amount of pollutant created, and scale this quantity by:

(1) the speed of pollutant dispersion under existing or predicted physical conditions,
(2) the severity of predicted pollutant levels from other sources, and
(3) the size of the population affected.

Solution Technique

The size of this problem, both in the number of variables and the number of constraint equations, represents a system which is too large for straightforward application of mixed integer, or sometimes even linear, programming techniques. The strategy is thus to set up the constraint equation matrix in a form as nearly diagonal as possible, using an ordering technique for the variables and constraints which has them progressing in time from one corner of the constraint matrix to the other. Efforts to block-diagonalize this matrix, such as with techniques that collect coupling terms at the ends of the matrix, destroy the very desirable time-sequential nature of this format. The solution strategy that is used is shown in Figure 2.

The integer decisions are "solved" over portions of time, in sequence, from the beginning to the end of the schedule period. These integer decision fields are overlapped slightly from one step to the next, and are imbedded within a linear program over the surrounding portion of the constraint matrix in which the other integer variables have either been fixed as previously decided, or they are relaxed to be continuous.

The linear program solution technique used is the Simplex-dual technique, and the binary decisions are made most quickly with a branch-and-bound search (as opposed to Gomory-cut techniques which are slower for binary integer problems). The dual variables available from this solution technique provide a number of valuable sensitivity measures including such factors as incremental costs of power, and closeness of decisions to the

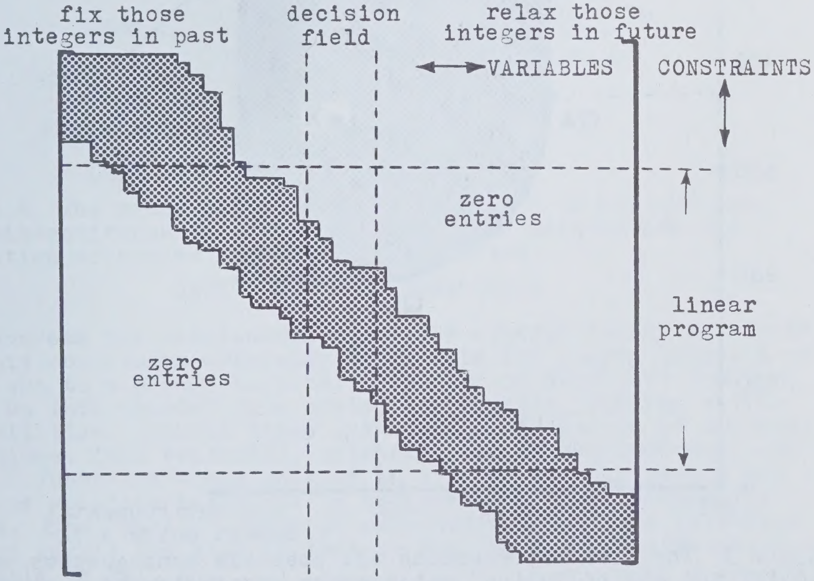


Figure 2 Representation of variables and constraint equations involved in an intermediate step in the sequential solution procedure.

basis (indicating branchpoints to near-optimal alternative paths). Although this technique will not always yield the pure optimum solution, this particular procedure has been found to be very desirable for a number of reasons:

(1) it decreases computation times considerably,
(2) it displays the optimal and the many near-optimal schedules over the various blocks of time,
(3) thus, it minimizes recomputation efforts when unforeseen system changes occur, and
(4) it keeps the system on a scheduling track around which a number of attractive alternative schedules exist.

Scheduling Example

A simple maintenance scheduling example is presented here (and has been presented in detail in Ref. 7). It involves a system of twelve power plants that are situated close to one another on the same river system. A number of intricate maintenance coordination problems are included.

Figure 3 displays all the "dollar versus air impact" scheduling options which are possible. Although any point in the shaded area is a feasible alternative, the dark line represents the set of all possible optimal solutions. The point labelled X represents the quality of a schedule which resulted from a common, nonobjective function, 'fill-in-the-valley' type of scheduling technique. The points QD and QA are the conse-

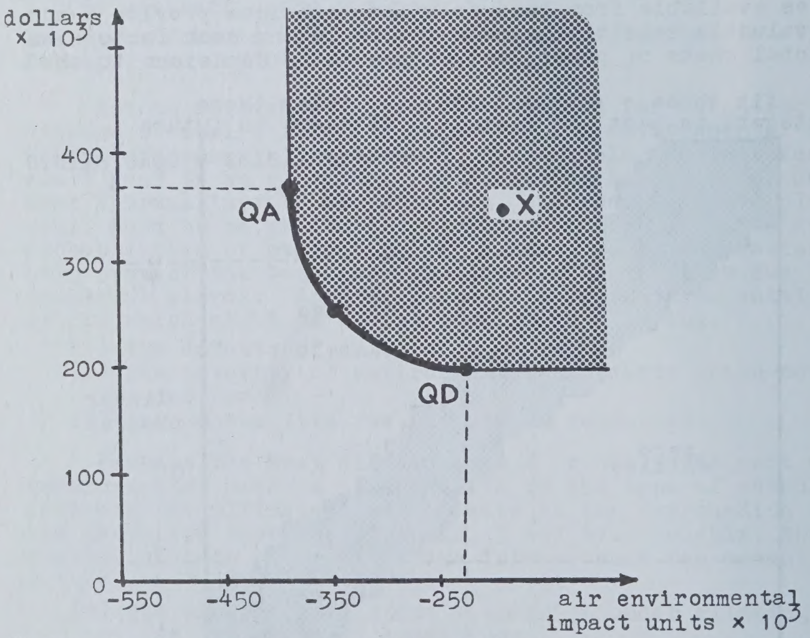


Figure 3 The area representing all possible consequences of "dollar and air pollution" maintenance strategies at a standard reliability level (the negative values on the air impact axis result from a convention of rewarding plants as a function of the time of year they are scheduled out for maintenance).

quences of the minimum dollar and the minimum air impact strategies. The shape of the dark curve is very flat near QD and QA showing that, for this hypothetical but representative system, it is wasteful to use a single measure of desirability. Adding the water impact dimension to this trade-off curve yields Fig. 4, which is then the surface of all optimum performance strategies of which this system is capable (QW is the minimum water impact point; QE, QV and QB are minimum air + water, water + dollar, and dollar + air points; and QT is the minimum of all three equally weighted).

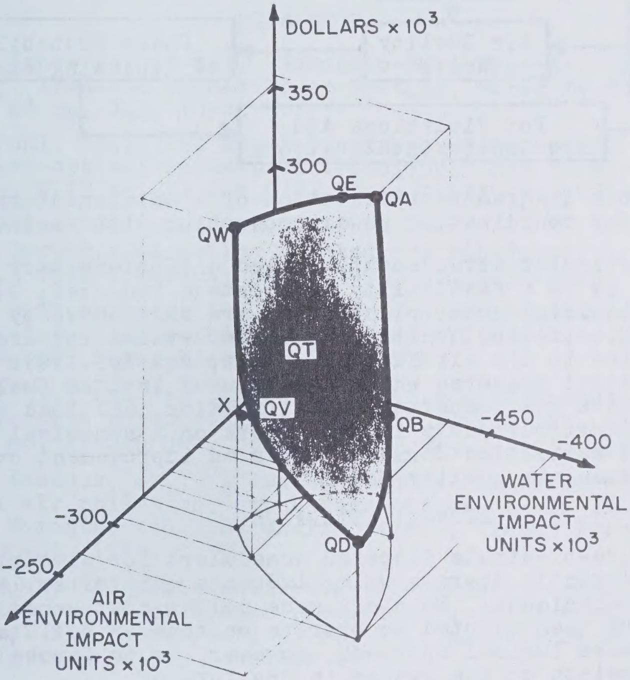


Figure 4 The trade-off surface associated with all optimum economic-environmental consequences of the maintenance and production schedules (standard reliability).

UNIT COMMITMENT SCHEDULING

Whereas the maintenance scheduler covered weekly intervals the unit commitment scheduler deals with the hourly dispatch of power out to a weekly horizon. A number of different problems arise on this shorter time scale, for example, loading curves of facilities, startup times and costs as functions of previous down times, fuel switching, spinning reserve requirements, and so on. These are again modeled in a mixed integer format and the same solution technique is employed for this large scale system. The "shield" shape of near-optimum possible solutions is very nearly the same as that in Fig. 4 (some results are available in Ref. 8).

In the maintenance schedule the only information on atmospheric dispersion potential is in the statistical, climatological data at each site. In the hourly scheduling problem, however, weather forecasts are available for the pollutant dispersion potential at each site for the next 24 to 36 hours. Rather than coupling this dispersive potential directly into the objective function, as is possible when using simple climatological data, a separate air quality model (in itself a large scale system) is being operated in conjunction with the unit commitment scheduler, as is shown in Figure 5.

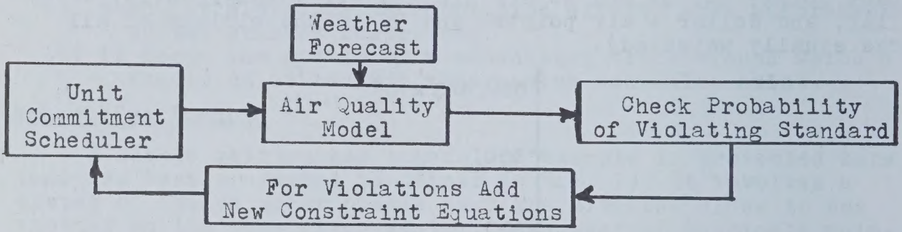

Weather Forecast
Unit Commitment Scheduler
Air Quality Model
Check Probability of Violating Standard
For Violations Add New Constraint Equations

Figure 5 Block diagram representation of a supplementary control system for coordinating power production with weather.

This particular arrangement, called a supplementary control system, is in a feasibility study stage (see Ref. 9) on a set of four isolated power plants that are surrounded by 17 air pollution concentration monitors (used for evaluating and feeding information to the air quality models, see Ref. 10). The pollution control measures which can be used include fuel switching, stack gas temperature modification, and load shifting. This effort currently includes work on a numerical diffusion, state estimation formulation as an improvement over the current Gaussian air quality model.

ECONOMIC DISPATCH

The minute-by-minute dispatch schedulers for electric power systems currently operate using LaGrange multiplier, incremental cost techniques. Some of these dollar cost schedulers (Ref. 11) have been adapted to operate on tons of pollutants emitted. A more logical approach, however, is to impose additional constraints on the system in the form of

$$\sum_{i=1}^{n} F_{ij}u_i \leqslant S \qquad (5)$$

where

F_{ij} = the contribution to the pollution concentration at point j from unit i being full on
u_i = the operating level of unit i
S = pollution concentration standard
i = the contributing pollution sources
j = the points which represent modes in pollution concentration violation areas.

These additional constraints (like those described in Ref. 12) are the same as those computed in the "feedback" loop in Fig. 5, and their use in the economic dispatch procedure ensures the compatibility of the operating objectives at these two different control levels.

CONCLUSION

Although it has not been possible to include most of the equations, the concepts have been presented that are being used in a large scale model of electric power operation and atmospheric dispersion potential. Unfortunately, even this rather large coordinated effort is just scratching the surface of the energy-environmental work which must be done, and particularly with the possibilities for food and resource 'crises' we will need more and more large scale system modeling efforts.

REFERENCES

1. F.C. Schweppe, et al., "Economic-Environmental System Planning," IEEE-PES Summer Power Meeting, Paper 74 CH0912-6-PWR, 28 pp, July 1974.
2. J. Gruhl, "Electric generation production scheduling using a quasi-optimal sequential technique," MIT Energy Lab, 153pp, NTIS no. PB-224 079, Springfield, Va. 22151, April 1972.
3. L.L. Garver, "Adjusting maintenance schedules to levelize risk," IEEE Transactions, PAS-91, pp. 2057-2063, Sep. 1972.
4. A.D. Patton and J. Ali, "Comparison of methods for generator maintenance scheduling," Paper C72 452-1, 1972 IEEE-PES Summer Power Meeting, July 1972.
5. J. Gruhl, "Quantification of aquatic environmental impact of electric power generation," MIT Energy Lab, 175 pp, NTIS no. PB-224 642, Springfield, Va., 22151, April 1973.
6. L.R. Babcock, Jr., "A combined index for measurement of total air pollution," in Determination of Air Quality, Gleb Mamantov and W.D. Shults (eds.), Plenum Press, N.Y., pp. 65-81, 1972.
7. J. Gruhl, "Economic-environmental-security transform curves of electric power system production schedules and simulations," MIT Energy Lab, 93 pp, NTIS no. PB-224 061, Springfield, Va., 22151, November 1972.
8. J. Gruhl, "Electric power unit commitment scheduling using a dynamically evolving mixed integer program," MIT Energy Lab, 150 pp, NTIS no. PB-224 006, Springfield, Va., 22151, January 1973.
9. M. Ruane, et al., "The Design and Implementation of a Demonstration Supplementary Control System," MIT Energy Lab, AEC Report COO-2428-2, 163 pp, September 1974.
10. B.A. Egan and J.P. Gaertner, "Modeling for Supplementary Control Systems," 5th Meeting of NATO/CCMS Expert Panel on Air Pollution Modeling, Riso, Denmark, 26 pp, June 1974.
11. J.W. Lamont and M.R. Gent, "Controlled-emission dispatch proposed," Electrical World, vol. 178, no. 2, pp. 60-61, July 15, 1972.
12. R.L. Sullivan, "Minimum pollution dispatching," IEEE-PES Summer Power Meeting, Paper C72-468-7, 4pp, July 1972.

WHEN IS A COMPLEX ECOSYSTEM STABLE?

D. D. Šiljak
School of Engineering
University of Santa Clara
Santa Clara, California

ABSTRACT

This work represents an attempt to formalize and partially resolve in the framework of Liapunov's stability theory the "complexity vs. stability" problem in model ecosystems. Sufficient (and sometimes necessary and sufficient) conditions are derived for stability of models under structural perturbations caused by time-varying nonlinear interactions among species in the community. The conditions guarantee that the community is stable despite arbitrary bounded variations in the interactions, which include the reduction of the number of species in the community, the changing of species from predator-prey to competitive interaction over the time interval, as well as interaction effects such as cutting trophic links between species, predator switching, and saturation of predator attack capacity.

Instability of ecomodels is formulated in the context of variable interactions among species. Sufficient conditions for instability are derived, which can help in producing the necessary condition for stability of the ecomodels.

Stability regions are also studied due to the fact that most of the useful models of population dynamics are not globally stable.

Hierarchic models of ecosystems are considered which describe communities decomposed into "blocks" of species representing various trophic levels or subcommunities. Stability of the total trophic web is determined on the basis of the aggregate model where each subcommunity is represented by a single variable, and which has the order equal to the number of subcommunities. The aggregate model represents not only a reduction of the dimensionality of the stability problems, but also provides an insight into essential structural properties of the interacting subcommunities.

I. INTRODUCTION

A central question in population ecology is: Does increased trophic web complexity lead to a higher degree of community stability, or is it the other way around and the complex communities are less stable than the simple ones? There is a good deal of experimental evidence collected by many people as surveyed recently by May in his comprehensive monograph [1], which supports both alternatives. The conclusive empirical evidence calls for attempts to answer the above question by theoretical stability studies of appropriate classes of model ecosystems. Such studies can make transparent the essential structural properties of observed communities in the context of complexity vs. stability problem, and provide certain both qualitative and quantitative results regarding the mechanism of diversity in multispecies communities.

In stability investigations of a linear model of n competing species, McArthur [2] formulated the complexity vs. stability problem in an important way as the effect on stability of increasing the number of species by one, i.e., to $n + 1$. By making the symmetry assumption in the model, he was able to use the well-known property of symmetric matrices and conclude that the $(n+1)$-species community is never more stable than the n-species community. The symmetry assumption, however, is overly restrictive and can hardly be used in stability studies of communities with more than one trophic level. In the spirit of McArthur's investigations, but otherwise using computer experiments, Gardner and Ashby [3] observed that increasing the level of interaction in a community of density dependent species, leads to a sudden instability regardless of the sign pattern in the interactions among species. These experiments were confirmed by May [4] for large numbers of species and a functional relationship was studied between the number of variables in the system and the average connectedness and interaction magnitude among the various variables. Although the experiments were useful in shedding more light on the complexity vs. stability problems, they suffered from the lack of mathematical generality and cannot be used to predict stability in new situations not actually tested.

In [5], it was claimed that the complexity vs. stability problem can be appropriately formulated in the framework of the general competitive-cooperative equilibrium analysis. The purpose of this work is to support the claim by a detailed mathematical analysis of stability in model ecosystems. We will show that McArthur's argument can be valid not only for asymmetrical linear constant models, but that it can be proved to hold in models with nonlinear time-varying interactions. Furthermore, we will be able to show that in competitive-cooperative model ecosystems, there is a definite level of complexity that can be always tolerated by stability. That level is determined on the basis of a dominant diagonal property of the constant community matrix, and can be carried over to nonlinear time-dependent models by using the Liapunov direct method. Such results provide a natural mathematical setting for support of the Gardner-Ashby and May experimental investigations.

The content of the paper is as follows: In the next section we will introduce the problem of complexity vs. stability in the linear constant models, and outline the solution with a minimum of mathematical technicalities. This will (hopefully) motivate the reader to continue on more complicated developments in the following sections. In Section 3, we will formulate connective stability [6, 7] using Liapunov's stability concept, which provides an explicit characterization of the system complexity. Then, we will provide necessary and sufficient conditions for that kind of stability in nonlinear models by the Liapunov's direct method. In Section 4, we will present further refinements of the obtained stability conditions for linear time-varying models. Section 5 is devoted to derivations of conditions for connective instability. Since most of the models in population ecology are not globally stable, in Section 6 we will give estimates of stability regions for nonlinear time-varying models. Section 7 considers hierarchic model ecosystems in which the species are organized in a number of interacting subcommunities (or trophic levels). No special sign pattern of interactions in the individual groups is assumed save that they produce a stable subcommunity. Then, the decomposition-aggregation stability analysis [6, 7] is used to show that stability of an aggregate matrix, the order of which is equal to the number of subcommunities (or trophic levels), implies stability of the entire community (or trophic web). Stability is again connective and subcommunities can be disconnected from each other and again connected together in various ways without affecting stability of the total community.

II. LINEAR CONSTANT MODELS

To answer some of the questions raised above and to set the stage for more elaborate analysis of nonlinear and time-varying effects in complex systems, let us consider the linear constant model of n interacting species described by the differential equation[1]

---

[1] With some obvious exceptions, lower case Roman letters denote vectors, capital Roman letters denote matrices, and Greek letters denote scalars.

$$\dot{x} = Ax , \tag{1}$$

where $x = x(t)$ is an n-vector $x = \{x_1 x_2 \ldots x_n\}$ the components of which represent the populations. The $n \times n$ constant community matrix $A = (a_{ij})$ has elements a_{ij} which characterize the influence of the population x_j of the j-th species on the population x_i of the i-th species in the community. We assume that all species are density dependent so that the matrix A has a negative diagonal, that is, $a_{ii} < 0$ for all $i = 1, 2, \ldots , n$. Such an assumption reflects the resource limitation in the community, and as pointed out by Tanner [8] is unlikely to be manifestly violated by any of the commonly studied populations but that of man! In the case of the off-diagonal elements a_{ij}, we do not specify their sign, thus allowing for a "mixed" competitive-predator-symbiotic-saprophytic interactions among species.

In order to study the effect of the strength of interactions among populations on the stability of the equilibrium $x_e = 0$ in the model (1), let us assume that the elements a_{ij} of the matrix A have the form

$$a_{ij} = \begin{cases} -\alpha_i + e_{ii}\alpha_{ii} , & i = j \\ e_{ij}\alpha_{ij} , & i \neq j . \end{cases} \tag{2}$$

The numbers e_{ij} can be considered as elements of the $n \times n$ constant *interconnection matrix* $E = (e_{ij})$, and can take any value between zero and one, that is

$$0 \leq e_{ij} \leq 1 . \tag{3}$$

for all $i, j = 1, 2, \ldots , n$. Therefore, by e_{ij} we can represent the influence of the population x_j on the population x_i ranging from a disconnection $(e_{ij} = 0)$ to a full connection $(e_{ij} = 1)$ measured by the number α_{ij} [7]. As for the diagonal elements a_{ii} of A, we retain the negative diagonal property by assuming that the numbers α_i and α_{ii} are related as $\alpha_i > \alpha_{ii} \geq 0$.

An important notion in the following stability analysis is that of the fundamental interconnection matrix $\bar{E}$ [7]. The matrix $\bar{E}$ is a constant interconnection matrix in which the elements $\bar{e}_{ij}$ take on binary values; 1 if the j-th population x_j influences the i-th population x_i; and 0 if x_j has no influence on x_i regardless of the changes in the community environment or the community itself. The matrix $\bar{E}$ is a binary matrix [9] which reflects the basic structure of the system. Therefore, any interconnection matrix E can be generated from the $\bar{E}$ by replacing the unit elements of $\bar{E}$ by corresponding elements e_{ij} of E.

Since the underlying characteristics of ecosystems are hardly ever known precisely and never fixed in time, it is necessary to assume possible perturbations in the interactions among the species in both the magnitude and direction. Influence of such perturbations on the stability properties of the system (1) were experimentally studied by Gardner and Ashby [3]. On the basis of a computer study, they concluded that if the strength of interconnections among the populations increases beyond a certain critical value, the system becomes unstable. This fact was later confirmed by May [4] who used it to draw conclusions about the relationship between the complexity of large systems expressed by connectedness among the variables and its stability properties.

Interconnection matrix represents a suitable means for describing the perturbations of interconnections and formulating the connective stability concept [6]: System (1) is connectively stable if it is stable for all interconnection matrices E generated by the fundamental interconnection matrix $\bar{E}$ of the system. In the context of connective stability, one can consider structural perturbations whereby certain species (or group of species) is disconnected from (and again connected with) the rest of the community. Such was the case with a community of marine invertebrates reported by Paine [10], where removal of one species caused the community to become unstable and shrink from fifteen-species to eight-species system. The structural perturbations of this kind can be readily described by interconnection matrices. Removal (or destruction) of the k-th species in the system (1) is represented by

$$\begin{aligned} e_{ik} &= 0 \ , \ i = 1, 2, \ldots, n \\ e_{kj} &= 0 \ , \ j = 1, 2, \ldots, n \ . \end{aligned} \tag{4}$$

If we show that the system (1) is connectively stable, then the structural perturbation (4) cannot produce instability of the equilibrium. Structural perturbations that change the number of species were analytically considered by MacArthur [2] assuming a special symmetric form of the community matrix A. Not only that we do not require this special symmetric form of A, but also we will wave the assumption of constancy of the system (1), thus allowing for time-varying structural perturbations which take place at disequilibrium populations and give rise to a time-dependent matrix $A = A(t)$.

Let us now present the conditions for the connective stability property of the system (1). We recall that an $n \times n$ matrix $A = (a_{ij})$ was called quasidominant diagonal by McKenzie [11] if there exist an n-vector $d = \{d_1 \ d_2 \ \ldots \ d_n\}$ with positive components $(d > 0)$ such that

$$d_j |a_{jj}| > \sum_{\substack{i=1 \\ i \neq j}}^{n} d_i |a_{ij}| \ , \ j = 1, 2, \ldots, n \ . \tag{5}$$

As shown by McKenzie, for a negative diagonal matrix A, inequalities (5) are sufficient for stability (all eigenvalues of A have negative real parts). That is, if the community matrix A is negative and quasidominant diagonal, the system (1) is globally asymptotically stable.

Condition (5) includes the usual diagonal dominance

$$|a_{jj}| > \sum_{\substack{i=1 \\ i \neq j}}^{n} |a_{ij}| \ , \ j = 1, 2, \ldots, n \tag{6}$$

as a special case when $d = \{1 \ 1 \ \ldots \ 1\}$. Although condition (6) is easier to interpret in the context of system (1), it is more restrictive than that of (5). It should be noted, however, that for large classes of nonlinear time-varying models we will be able to replace condition (5) with readily interpretable inequalities involving only a_{ij}'s.

Now, if we establish (5) for a given $\bar{E}$, then it will be satisfied for any E generated by that $\bar{E}$. To see this, we use (2) to rewrite (5) as

$$d_j |-\alpha_j + e_{jj}\alpha_{jj}| > \sum_{\substack{i=1 \\ i \neq j}}^{n} d_i |e_{ij}\alpha_{ij}| \ , \ j = 1, 2, \ldots, n \ . \tag{7}$$

Obviously, if (7) holds for $\bar{E}$ it will hold for any E which is obtained from $\bar{E}$ by replacing unit elements of $\bar{E}$ by the numbers e_{ij} which lie between zero and one as defined in (3). Specifically, (7) will hold in case of the structural perturbations (4), and removal of the k-th species from a stable competitive community described by (1) will not cause instability.

From the above analysis, it is possible to draw conclusions about the tolerance of stability to increasing complexity of the system: So long as the strength of interactions does not violate inequalities (5), stability of the community is assured. This conclusion is compatible with the empirical evidence noticed by Margalef [12] "... it seems that species that interact feebly with others do so with a great number of other species. Conversely, species with strong interactions are often part of a system with a small number of species ...". and it does not contradict the statement of Levins [13] that "there is often a limit to the complexity of systems".

Since the conditions (5) are only sufficient for stability, they cannot be used to test instability and verify entirely the Gardner-Ashby [3] and May [4] experiments. Conditions (5), however, are both necessary and sufficient for stability of (1) if the interactions obey the rules "friends of friends are friends, enemies of enemies are friends, and friends of enemies and enemies of friends are enemies". This situation is important in the analysis of substitute-complement multiple markets of commodities of services and is known as the Morishima case [14]. In the context of the community model (1), it means that the community is constituted of species in either competitive or symbiotic mutual relationships according to the rules given above. A special case of the Morishima matrix A is that of the Metzler matrix used in economics [14] to represent the case when all commodities are substitutes. A community model (1) described by a Metzler matrix is of little significance in population dynamics since it represents a rare "gross-mutualism" among species. It should be noted, however, that in case of the Metzler matrix A, conditions (5) are necessary and sufficient for stability and if violated, the community would suddenly become unstable. This conclusion certainly agrees with the May's speculation [1] that "predator-prey bonds are more common than symbiotic ones", and via the Morishima matrix the conclusion can be extended to his statement that "competition or mutualism between two species is less conductive to overall web stability than is a predator-prey relationship".

Since diagonal dominance in A includes the case of complete disconnection of all the species when $E = 0$, we need the negative diagonal of A for stability. Therefore, we have stability within each trophic level and that implies stability of the entire trophic web if (and only if) the strength of interconnections is such that inequalities (5) are satisfied. With regard to Holling's conjecture [1] that stability on any of the single trophic levels enforces stability of the web, we can say that web stability implies stability of each individual trophic level if the community matrix is quasidominant diagonal.

Finally, in the context of the model (1) we should mention that if the community matrix A is negative diagonal then the conditions (5) imply that it is D-stable [14], that is, the product DA is stable for any positive diagonal matrix $D = \text{diag}\,\{d_{11}, d_{22}, \ldots, d_{nn}\}$ in which $d_{ii} > 0$ for all i and $d_{ij} = 0$ for all $i \neq j$. This fact shows a high degree of reliability of stability in communities described by negative quasidominant diagonal matrices, and that they are stable under arbitrary "speeds of adjustments" of populations in the community. Unfortunately, neither D-stability nor closely related qualitative stability which was applied to population dynamics by May [1], can be used for other than linear constant models. This is not true, however, in the case of the connective stability concept introduced above, which can be used to establish stability of nonlinear time-varying models at the expense of more refined analysis presented in the following section.

III. NONLINEAR MODELS

Interactions in multispecies communities is a highly nonlinear and nonstationary affair, and linear constant models represent the simplest approximation of the true situation. There are several ways to bring the stability analysis of the preceeding section closer to reality, and a step in that direction is to consider the nonlinear time-varying model described by the differential equation

$$\dot{x} = A(t, x)x , \tag{8}$$

which is an obvious generalization of the linear constant model

$$\dot{x} = \bar{A}x , \tag{9}$$

in which the constant elements $\bar{a}_{ij}$ of the $n \times n$ matrix $\bar{A} = (\bar{a}_{ij})$ are replaced by nonlinear time-dependent functions

$$a_{ij} = a_{ij}(t, x) . \tag{10}$$

Equation (9) is that of (1) and the bar above $\bar{A}$ is used to distinguish it from $A(t, x)$ since we will omit the arguments (t, x) of A for simplicity.

Stability analysis of the nonlinear model (8) cannot be carried out in algebraic terms as it is possible to do in the case of the linear model (9), and we need to use the Liapunov direct method from the theory of ordinary differential equations [15]. It is, however, of fundamental importance to note that we will be able to imply various stability properties of the nonlinear time-varying model (8) from the stability properties of the linear constant model (9) established by simple algebraic criteria involving only the elements $\bar{a}_{ij}$ of the matrix $\bar{A}$ in (9).

In (8), $x(t) \in \mathcal{R}^n$ is the population vector, and the $n \times n$ matrix function $A\colon \mathcal{T} \times \mathcal{R}^n \to \mathcal{R}^{n^2}$ is defined, continuous and bounded on $\mathcal{T} \times \mathcal{R}^n$ so that the solutions $x(t; t_0, x_0)$ of (8) exist for all initial conditions $(t_0, x_0) \in \mathcal{T} \times \mathcal{R}^n$ and $t \in \mathcal{T}_0$. The symbol $\mathcal{T}$ represents the time interval $(\tau, +\infty)$, where τ is a number or the symbol $-\infty$, and $\mathcal{T}_0$ is the semi-infinite time interval $[t_0, +\infty)$.

In the following analysis, we will consider stability of the equilibrium population $x_e = 0$ of the model (8). If $A(t, x_e)x_e = 0$, $\forall t \in \mathcal{T}$, and $x_e \neq 0$ is of interest, then we can define the nonlinear matrix function $B(t, y)y \equiv A(t, y + x_e)(y + x_e)$ and consider the equation $\dot{y} = B(t, y)y$ instead of (8), where $y_e = 0$ represents the equilibrium x_e under investigation. Although it is always possible in principle to use the transformation $x = y + x_e$ and place the equilibrium at the orgin of $\mathcal{R}^n$, the new transformed equation is usually more cumbersome to handle.

In order to include the connective property of stability in the analysis, we write the elements $a_{ij}(t, x)$ of $A(t, x)$,

$$a_{ij}(t, x) = \begin{cases} -\varphi_i(t, x) + e_{ii}(t)\, \varphi_{ii}(t, x) , & i = j \\ e_{ij}(t)\, \varphi_{ij}(t, x) , & i \neq j \end{cases} \tag{11}$$

where the functions $\varphi_i(t, x), \varphi_{ij}(t, x) \in C^{(0,0)}(\mathcal{T} \times \mathcal{R}^n)$.

In (11), we consider the elements $e_{ij} = e_{ij}(t)$ of the $n \times n$ interconnection matrix $E = (e_{ij})$ as functions of time such that $e_{ij}(t) \in C^0(\mathcal{T})$ and restricted only by the condition $e_{ij}(t) \in [0,1]$. This represents a major improvement over the use of E in the previous section since structural perturbations can take place at the disequilibrium populations without any prescribed pattern - the elements $e_{ij}(t)$ may be arbitrary functions of time without any statistical description. This is remarkable in that we allow the interactions among the species in a community to vary arbitrarily in strength during the population adjustment process without any a priori specification of such variations save that they are bounded.

Now, we state the definition of connective stability introduced in [6]:

Definition 1: The equilibrium $x_e = 0$ of the system (8) is connectively stable if and only if it is stable in the sense of Liapunov for all interconnection matrices E.

To derive necessary and sufficient conditions for connective stability, we need further specifications of the matrix elements $a_{ij}(t, x)$. We assume that the functions $\varphi_i(t, x)$ and $\varphi_{ij}(t, x)$ in (11) are bounded functions on $T \times R^n$ and that there exist positive numbers $\alpha_i > 0$, $\alpha_{ij} \geq 0$ such that

$$\alpha_i = \inf_{T \times R^n} \varphi_i(t, x), \quad \alpha_{ij} = \sup_{T \times R^n} |\varphi_{ij}(t, x)| \qquad \forall i, j = 1, 2, \ldots, n \tag{12}$$

and $\alpha_i > \alpha_{ii}$.

In deriving the stability conditions, we can improve Definition 1 in two important ways:

First, we can establish connective stability of (8) for any set of functions $\varphi_i(t, x)$, $\varphi_{ij}(t, x)$ satisfying conditions (12), that is, for functions that belong to the following classes of functions:

$$\begin{aligned} \Phi_i &= \{\varphi_i(t, x): \varphi_i(t, x) \geq \alpha_i\} \\ \Phi_{ij} &= \varphi_{ij}(t, x): |\varphi_{ij}(t, x)| \leq \alpha_{ij}\}, \end{aligned} \tag{13}$$

$\alpha_{ij} \geq 0$, $\alpha_i > \alpha_{ii}$. Similarly as in the case of the elements of the interconnection matrix, we do not have to specify the precise shape of the nonlinear interactions among the species in the community so long as the interactions are bounded. This is quite a realistic assumption since most of the familiar models of ecosystems surveyed by May [1] have this property. In particular, restrictions (13) include the case of saturation characteristics (e.g., saturation of predator attack capacities, death rate among predators and prey) as well as predator switching, nonlinearity in the food supply, etc. Furthermore, conditions (12) include any sign pattern of the interactions, thus allowing for "mixed" (competitive-predator-symbiotic-saprophytic) communities.

Second improvement of Definition 1 will be in the ability of the analysis to estimate how fast the populations tend to the equilibrium values, thus providing a significant quantitative aspect of community stability. In fact, we will provide necessary and sufficient conditions for stability of (8) described by the following definition.

Definition 2: The equilibrium $x_e = 0$ of the system (8) is absolutely, exponentially, and connectively stable if and only if there exist two positive numbers Π and π independent of initial conditions (t_0, x_0) such that

$$||x(t; t_0, x_0)|| \leq \Pi ||x_0|| \exp[-\pi(t-t_0)], \quad \forall t \in T_0 \tag{14}$$

for all $(t_0, x_0) \in T \times R^n$, all $\varphi_i \in \Phi_i$, $\varphi_{ij} \in \Phi_{ij}$, and all interconnection matrices E.

In Definition 2, the term "absolute" refers to the fact that stability is established not for specific functions $\varphi_i(t, x)$, $\varphi_{ij}(t, x)$, but for the classes of functions Φ_i, Φ_{ij}. The term "exponential" is used to acknowledge explicitly inequality (14) which ensures that the population vector $x(t)$ tends to the equilibrium $x_e = 0$ faster than an exponential.

To establish the kind of stability specified by Definition 2, we denote by $\bar{A} = (\bar{a}_{ij})$ the constant $n \times n$ matrix with elements

$$\bar{a}_{ij} = \begin{cases} -\alpha_i + \bar{e}_{ii}\alpha_{ii}, & i = j \\ \bar{e}_{ij}\alpha_{ij}, & i \neq j \end{cases} \tag{15}$$

where $\bar{e}_{ij}$'s take on values 1 or 0 according to the fundamental interconnection matrix $\bar{E}$. Since the elements $\bar{a}_{ij}$ of $\bar{A}$ are such that

$$\bar{a}_{ij} \begin{cases} < 0, & i = j \\ \geq 0, & i \neq j \end{cases} \tag{16}$$

$\bar{A}$ is a Matzler matrix regardless of the sign of interactions in the original nonlinear matrix function $A(t, x)$. A Metzler matrix $\bar{A}$ is stable if and only if it is quasidominant diagonal [11] or, equivalently, is a Hicks matrix, that is, the sign of its q-th order principal minor is $(-1)^q$. This is a well known result in economics [14], and it is surprising to learn that the economics literature does not acknowledge the fact that for Metzler matrices, the Hicksian property can be reduced to the Sevastyanov-Kotelyanskii inequalities [16],

$$\bar{a}_{11} < 0, \quad \begin{vmatrix} \bar{a}_{11} & \bar{a}_{12} \\ \bar{a}_{21} & \bar{a}_{22} \end{vmatrix} > 0, \ \ldots, \ (-1)^n \begin{vmatrix} \bar{a}_{11} & \bar{a}_{12} & \ldots & \bar{a}_{1n} \\ \bar{a}_{21} & \bar{a}_{22} & \ldots & \bar{a}_{2n} \\ \ldots & \ldots & \ldots & \ldots \\ \bar{a}_{n1} & \bar{a}_{n2} & \ldots & \bar{a}_{nn} \end{vmatrix} > 0. \tag{17}$$

whereby only the sign of leading principle minors of $\bar{A}$ need be tested for stability.

Now we prove the following:

Theorem 1. The equilibrium $x_e = 0$ of the system (8) is absolutely, exponentially, and connectively stable if and only if the corresponding $n \times n$ constant Metzler matrix $\bar{A}$ defined in (15) is a quasidominant diagonal matrix.

Proof. To prove the "if" part of Theorem 1, let us consider a decrescent, positive definite, and radially unbounded function $\nu: R^n \to R_+$,

$$\nu(x) = \sum_{i=1}^{n} d_i |x_i| \tag{18}$$

as a candidate for Liapunov's function which was introduced by Rosenbrock [17]. In (18), $d = (d_1\ d_2\ \ldots\ d_n)^T$ is a constant yet unspecified n vector with positive components $(d > 0)$.

We take the total time derivative $D^+\nu[x(t)]$ of the function $\nu[x(t)]$ along the solutions of (8). Since the derivative of $|x_i(t)|$ need not exist at a point where $x_i(t) = 0$, it is necessary to use the right-hand derivative $D^+|x_i(t)|$ as proposed by Rosenbrock [17] and later by Sandberg [18]. For this purpose, a functional σ_i is defined as

$$\sigma_i = \begin{cases} 1, \text{ if } x_i(t) > 0 \text{ , or if } x_i(t) = 0 \text{ and } \dot{x}_i(t) > 0 \\ 0, \text{ if } x_i(t) = 0 \text{ and } \dot{x}_i = 0 \\ -1, \text{ if } x_i(t) < 0 \text{ , or if } x_i(t) = 0 \text{ and } \dot{x}_i(t) < 0 \text{ ,} \end{cases} \tag{19}$$

where $x_i(t) \in C^1(T)$. Then,

$$\begin{aligned} D^+\nu(x) &= \sum_{i=1}^{n} d_i\sigma_i\dot{x}_i \\ &= \sum_{i=1}^{n} d_i\sigma_i \sum_{j=1}^{n} a_{ij}x_j \\ &= \sum_{j=1}^{n} d_j\sigma_j x_j a_{jj} + \sum_{j=1}^{n} x_j \sum_{\substack{i=1 \\ i\neq j}}^{n} d_i\sigma_i a_{ij} \\ &\leq -\sum_{j=1}^{n} d_j|x_j||\bar{a}_{jj}| + \sum_{j=1}^{n} |x_j| \sum_{\substack{i=1 \\ i\neq j}}^{n} d_i|\bar{a}_{ij}| \text{ .} \end{aligned} \tag{20}$$

Since $\bar{A}$ is quasidominant diagonal, there exists a vector $d > 0$ and a number $\pi > 0$ such that

$$|\bar{a}_{jj}| - d_j^{-1} \sum_{\substack{i=1 \\ i\neq j}}^{n} d_i|\bar{a}_{ij}| \geq \pi \text{ , } \forall j = 1, 2, \ldots , n \text{ .} \tag{21}$$

From (20) and (21), we get the differential inequality

$$D^+\nu \leq \pi\nu \text{ , } \forall t \in T_0 \text{ , } \forall \nu \in R_+ \tag{22}$$

valid $\forall\varphi_i(t, x) \in \Phi_i$, $\forall\varphi_{ij}(t, x) \in \Phi_{ij}$, $\forall E$. By integrating (22) we obtain

$$\nu[x(t)] \leq \nu(x_0) \exp[-\pi(t-t_0)] \text{ ,} \qquad \forall t \in T_0 \text{ , } \forall(t_0, x_0) \in T \times R^n \text{ .} \tag{23}$$

Using the well-known relationship between the Euclidean and absolute-value norms $||x|| \leq |x| \leq n^{1/2} ||x||$, we can rewrite (23) as

$$\begin{aligned} &||x(t; t_0, x_0)|| \leq \Pi||x_0|| \exp[-\pi(t-t_0)] \text{ , } \forall t \in T_0 \\ &\forall(t_0, x_0) \in T \times R^n \text{ , } \forall\varphi_i \in \Phi_i \text{ , } \forall\varphi_{ij} \in \Phi_{ij} \text{ , } \forall E \end{aligned} \tag{24}$$

with

$$\Pi = n^{1/2} d_M d_m^{-1} \text{ ,} \tag{25}$$

where $d_M = \max_i d_i$, $d_m = \min_i d_i$. This proves the "if" part of Theorem 1.

For the "only if" part of Theorem 1, we select the particular system (8) specified by

$$\varphi_i(t, x) = \alpha_i \text{ , } \varphi_{ij}(t, x) = \alpha_{ij} \qquad \forall i, j = 1, 2, \ldots , n \tag{26}$$

and the fundamental interconnection matrix $\bar{E}$, that is, the matrix $A(t, x)$ in (8) is the constant Metzler matrix $\bar{A}$. If the matrix $\bar{A}$ is not quasidominant diagonal, then the chosen system is unstable for $E = \bar{E}$, and the system (8) is not connectively stable for $\varphi_i(t, x) = \alpha_i \in \Phi_i$, $\varphi_{ij}(t, x) = \alpha_{ij} \in \Phi_{ij}$. This proves Theorem 1.

The basic result stated in Theorem 1 appears in different forms in the work of Rosenbrock [17], Persidskii [19], Bellman [20], and for linear time-varying models in the work of Sandberg [18]. Theorem 1 can be considerably generalized to include a wider class of interactions at the expense of establishing weaker asymptotic properties of nonlinear matrix models.

It is of interest to interpret the result of Theorem 1 in the ecological context. Let us immediately notice that despite the fact that Theorem 1 provides the necessary and sufficient conditions for stability, they are so in the sense of absolute stability. For any particular choice of the functions $\varphi_i(t, x)$, $\varphi_{ij}(t, x)$, the conditions are only sufficient. Still, the result is compatible with Gardner-Ashby's [3], and May's [4] experiments: In varying the values of the off-diagonal elements of the community matrix, one has to violate conditions (17) or, which is the same, conditions (21) in order to produce instability. Furthermore, according to Theorem 1, it is irrelevant how one changes the interactions. Stability would prevail as long as the constraints (12) and inequalities (21) are satisfied.

IV. LINEAR TIME-VARYING MODELS

It is possible to establish necessity and sufficiency of diagonal dominance for stability in a special case of the linear time-varying model

$$\dot{x} = A(t)x , \tag{27}$$

where the $n \times n$ matrix $A(t) = [a_{ij}(t)]$ has time-varying elements $a_{ij}(t) \in C^0(\mathcal{T})$ defined by

$$a_{ij} = \begin{cases} -\alpha_i + e_{ii}(t)\,\alpha_{ii} , & i = j \\ e_{ij}(t)\,\alpha_{ij} , & i \neq j . \end{cases} \tag{28}$$

In (28), $\alpha_i > 0$ and α_{ij} are numbers such that $\alpha_i > \alpha_{ii}$. That is, the requirement placed on the signs of $a_{ij}(t)$ in $A(t)$ of (27) is, $a_{ii}(t) < 0$, $\forall t \in \mathcal{T}$, $\forall E$, and $A(t)$ is negative diagonal matrix for all t. In addition, we assume that more structural information about the system (27) is known and that the sign pattern in A follows the Morishima case mentioned in Section 2.

We introduce here the notion of a *time-varying Morishima matrix* as a matrix $C(t)$ which is indecomposable[2] for all $t \in \mathcal{T}$ and by identical row and column transpositions can be brought into the form

$$C(t) = \begin{bmatrix} C_{11}(t) & C_{12}(t) \\ C_{21}(t) & C_{22}(t) \end{bmatrix} , \tag{29}$$

where the square submatrices $C_{11}(t) \geq 0$, $C_{22}(t) \geq 0$ are nonnegative and the submatrices $C_{12}(t) \leq 0$, $C_{21}(t) \leq 0$ are nonpositive for all $t \in \mathcal{T}$.

We consider the model (27) when

$$A(t) = C(t) - D , \tag{30}$$

where $C(t) = [e_{ij}(t)\,\alpha_{ij}]$ is a time-dependent $n \times n$ Morishima matrix and

---

[2] A matrix A is said to be indecomposable if it cannot be permuted into a matrix of the form $\begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}$, where A_{11}, A_{22} are square submatrices and 0 is a block of zeros [14].

$D = \text{diag}\,\{\alpha_1, \alpha_2, \dots, \alpha_n\}$.

Since the elements a_{ij} of A are not functions of x, we ignore the absolute property of stability, and derive conditions for exponential connective stability of system (27). To do that, we denote by $\bar{A} = (\bar{a}_{ij})$ the matrix $A(t)$ which corresponds to $\bar{E}$,

$$\bar{a}_{ij} = \begin{cases} -\alpha_i + \bar{e}_{ii}\alpha_{ii}\,, & i = j \\ \bar{e}_{ij}\,\alpha_{ij}\,, & i \neq j \end{cases} \tag{31}$$

We prove the following:

Theorem 2. The equilibrium $x_e = 0$ of the system (27) is exponentially connectively stable in the large if and only if the $n \times n$ constant matrix $\bar{A}$ defined by (31), is a quasidominant diagonal matrix.

Proof. The "if" part of Theorem 2 follows from Theorem 1.

In order to show the "only if" part of Theorem 2, we notice from [14] that a negative diagonal matrix $\bar{A} = \bar{C} - D$ is stable if and only if $\bar{A}$ is quasidominant diagonal since $\bar{C}$ (which corresponds to $\bar{E}$) is a constant Morishima matrix. Therefore, violation of the quasidominant diagonal property of $\bar{A}$ implies that $\bar{A}$ is not a stable matrix and $x_e = 0$ is not stable for $E = \bar{E}$. The proof of Theorem 2 is complete.

Metzler matrix can be obtained as a special case of the matrix A in (30) by assuming that C_{22} in (29) is empty and $C(t) \equiv C_{11}(t)$. Therefore, if the community matrix A in (27) is a time-varying Metzler matrix, Theorem 2 provides necessary and sufficient condition for stability of the community model (27). This situation is of little interest, however, since it describes a community of species with "gross symbiotic" interactions.

V. INSTABILITY

Instability analysis of population dynamics appears in connection with the problem of stability at one trophic level versus web stability, and so far is on a conjecture-experiment level as reported by May [1]. Two basic questions are: (1) If one particular trophic level is unstable, can the total community be stable? (Paine's conjecture [10]); and (2) if one level in isolation is stable, can the total system be unstable? (Watt's conjecture [22]). As May pointed out, by judicious choice of interactions, both situations are possible. In this section, we will not try to answer either of the two questions, but rather determine conditions under which instability of species when isolated implied community instability. Such a result, however, provides a partial answer to both of the above questions, and sets a rigorous basis for further exploration of these questions in the multi-level stability analysis presented in the second part of this paper.

In proving Theorems 1 and 2, we implicitly used the notion of connective instability. The system (8) was said to be connectively unstable if it was not stable in the sense of Liapunov for at least one interconnection matrix. This is a natural instability counterpart to Definition 1 and can be resolved using the results of [23]. Since Theorems 1 and 2 provide necessary and sufficient conditions for stability of (8), they can be used for testing instability in the context of structural perturbations. Therefore, we are not interested in pursuing such investigations, but rather show that the mechanism used to prove connective stability can also be used to prove connective instability for all interconnection matrices - this is *complete connective instability*. In the ecological context, it means that no matter how the direction and strength of interactions may vary, the community remains unstable. More precisely, we state the following:

Definition 3. The equilibrium $x_e = 0$ of the system (8) is completely connectively unstable if and only if it is unstable in the sense of Liapunov for all interconnection matrices E .

Moreover, we can show that the connective instability is also exponential as in the following:

Definition 4. The equilibrium $x_e = 0$ of the system (8) is completely and exponentially connectively unstable in the large if and only if there exist two positive numbers Π and π independent of the initial conditions (t_0, x_0) such that

$$||x(t; t_0, x_0)|| \geq \Pi ||x_0|| \exp [\pi(t-t_0)] , \quad \forall t \in \mathcal{T}_0 \tag{32}$$

for all $(t_0, x_0) \in \mathcal{T} \times \mathcal{R}^n$ and all interconnection matrices E .

Let us assume that the coefficients $a_{ij}(t, x)$ of the system matrix $A(t, x)$ in (8) are defined by

$$a_{ij}(t, x) = \begin{cases} \varphi_i(t, x) - e_{ii}(t)\, \varphi_{ii}(t, x), & i = j \\ -e_{ij}(t)\, \varphi_{ij}(t, x), & i \neq j \end{cases} \tag{33}$$

and that there exist numbers $\alpha_i > 0$, $\alpha_{ij} \geq 0$, such that $\alpha_i > \alpha_{ii}$ and inequalities (12) are satisfied by the functions $\varphi_i(t, x)$, $\varphi_{ij}(t, x)$. Then, we specify the elements $\bar{a}_{ij}$ of the matrix $\bar{A} = (\bar{a}_{ij})$ by $\delta_{ij}\alpha_i - \bar{e}_{ij}\alpha_{ij}$ and the corresponding matrix $\bar{E}$.

We prove the following:

Theorem 3. The equilibrium state $x_e = 0$ of the system (8) is completely connectively and exponentially unstable in the large if the $n \times n$ constant Metzler matrix $-\bar{A} = (-\bar{a}_{ij})$ is a quasidominant diagonal matrix.

Proof. Let us again consider the function $\nu(x)$ of (18). Taking the total time-derivative of $\nu[x(t)]$ along solutions of (8) and using (33), we get

$$\begin{aligned} D^+\nu(x) &= \sum_{i=1}^{n} d_i \sigma_i \dot{x}_i \\ &= \sum_{i=1}^{n} d_i \sigma_i \sum_{j=1}^{n} a_{ij} x_j \\ &= \sum_{j=1}^{n} d_j \sigma_j x_j a_{jj} + \sum_{j=1}^{n} x_j \sum_{\substack{i=1 \\ i\neq j}}^{n} d_i \sigma_i a_{ij} \\ &\geq \sum_{j=1}^{n} d_j |x_j| \bar{a}_{jj} - \sum_{j=1}^{n} |x_j| \sum_{\substack{i=1 \\ i\neq j}}^{n} d_i |\bar{a}_{ij}| \end{aligned} \tag{34}$$

Since $-\bar{A}$ is a Metzler matrix and has a quasidominant diagonal, there wxists a vector $d > 0$ and a number $\pi > 0$ such that

$$\bar{a}_{jj} - d_j^{-1} \sum_{\substack{i=1 \\ i\neq j}}^{n} d_i |\bar{a}_{ij}| \geq \pi , \quad \forall j = 1, 2, \ldots , n . \tag{35}$$

Therefore, from (34) and (35) we get

$$D^+\nu \geq \pi\nu \,, \qquad \forall t \in \mathcal{T}_\alpha \,, \ \forall \nu \in \mathcal{R}_+ \,, \ \forall E \,. \tag{36}$$

Integrating (36) and manipulating the result similarly as (22), we get inequality (32) with

$$\Pi = n^{-1/2} d_m d_M^{-1} \,, \tag{37}$$

where $d_m = \min_i d_i$, $d_M = \max_i d_i$. This completes the proof of Theorem 3.

It is of interest to note that the quasidominant diagonal property of the matrix $-\bar{A}$ can be a necessary and sufficient condition if instability in Theorem 3 is required to be absolute as in Theorem 1. If instability is going to hold for all $\varphi_i(t, x) \in \Phi_i$, $\varphi_{ij}(t, x) \in \Phi_{ij}$, it must hold also for $\varphi_i(t, x) = \alpha_i$, $\varphi_{ij}(t, x) = -\alpha_{ij}$ which makes the system (8) with $A(t, x) = \bar{A}$ unstable since $-\bar{A}$ is a quasidominant diagonal matrix.

Theorem 3 has a rather weak application to population models since it implies that none of the species in the community is density dependent. That is, the diagonal elements $a_{ii}(t, x)$ of the matrix $A(t, x)$ are all positive.

There is a possibility to relax positivity conditions placed on all a_{ii}'s and require that only one of the populations is not density dependent. Then, however, we require positivity of all off-diagonal coefficients a_{ij} for that population. More precisely, we have the following:

Theorem 4. The equilibrium $x_e = 0$ of the system (8) is completely connectively unstable if for some $i \in \{1, 2, \ldots, n\}$ there exist numbers $\alpha_i > 0$, $\alpha_{ij} \geq 0$ such that the coefficients $a_{ij}(t, x)$ of the $n \times n$ matrix $A(t, x)$ defined by (26) satisfy the conditions

$$\varphi_i(t, x)x_i \geq \alpha_i x_i \,, \ \varphi_{ij}(t, x)x_j \leq -\alpha_{ij}|x_j| \qquad \forall t \in \mathcal{T} \,, \ \forall x \in \Delta \tag{38}$$

where $\Delta = \{x \in \mathcal{R}^n \colon ||x|| < \rho\}$ and $\rho > 0$.

Proof. The proof of Theorem 4 is almost automatic. Using (38), the i-th state equation can be rewritten as

$$\begin{aligned} \dot{x}_i &= \varphi_i(t, x)x_i - \sum_{j=1}^{n} e_{ij}(t)\, \varphi_{ij}(t, x)x_j \\ &\geq \alpha_i x_i + \sum_{j=1}^{n} e_{ij}(t)\, \alpha_{ij}|x_j| \\ &\geq \alpha_i x_i \,, \qquad \forall (t, x) \in \mathcal{T} \times \Delta \,, \ \forall E \,. \end{aligned} \tag{39}$$

By integrating the last inequality in (39), we get

$$|x_i(t; t_0, x_0)| \geq x_{i0} \exp\,[\alpha_i(t-t_0)] \,, \qquad \forall t \in \mathcal{T}_0 \,, \ \forall (t_0, x_0) \in \mathcal{T} \times \Delta \,, \ \forall E \,. \tag{40}$$

where $x_{i0} = x_i(t_0) > 0$. Inequality (40) implies complete connective instability of the equilibrium $x_e = 0$ in (8) and the proof of Theorem 4 is completed.

VI. STABILITY REGIONS

In most predator-prey models ranging from the original one of Lotka-

Voltera to the models examined by Ayala et. al. [24] and May [1], there exist more than one equilibria. Multiple equilibria rule out global stability and stability (if present) is restricted on a finite region of the population space. It is the purpose of this section to show how the theory of preceeding sections can be used to estimate regions of connective stability of nonlinear time-varying model described by equation (8).

We start with the following:

Definition 5. A region $\mathcal{A} \subset \mathcal{R}^n$ is a region of connective asymptotic stability associated with the equilibrium $x_e = 0 \in \mathcal{A}$ of the system (8) if and only if for all interconnection matrices E corresponding to $\bar{E}$, $x_e = 0$ is stable in the sense of Liapunov, and for $t \in \mathcal{T}_0$, $\lim_{t\to+\infty} x(t;\, t_0,\, x_0) = 0$ whenever $x_0 \in \mathcal{A}$.

Let us assume that nonlinear constraints (12) are satisfied only on a finite region $\mathcal{B}$ of the state space $\mathcal{R}^n$,

$$\alpha_i = \inf_{\mathcal{T}\times\mathcal{R}^n} \varphi_i(t,\, x),\quad \alpha_{ij} = \sup_{\mathcal{T}\times\mathcal{R}^n} |\varphi_{ij}(t,\, x)|$$
$$\forall i,\, j = 1,\, 2,\, \ldots,\, n;\ \forall (t,\, x) \in \mathcal{T}\times\mathcal{B} \tag{41}$$

for some numbers $\alpha_{ij} \geq 0$, $\alpha_i > \alpha_{ii}$, and in some region $\mathcal{B} \subset \mathcal{R}^n$ defined by

$$\mathcal{B} = \{x\colon |x_i| < \beta_i,\ \forall i = 1,\, 2,\, \ldots,\, n\} \tag{42}$$

and numbers $\beta_i > 0$.

We consider again the function $\nu(x)$ in (18), and assume that the matrix $\bar{A}$ with elements $\bar{a}_{ij}$ defined by (12) is quasidominant diagonal.

A region $\mathcal{A}$ of connective asymptotic stability is given by the following:

Theorem 5. The region $\mathcal{A}$ defined by

$$\mathcal{A} = \{x\colon \nu(x) < \min_i d_i\beta_i\} \tag{43}$$

is a region of connective asymptotic stability corresponding to $x_e = 0$ and $\bar{E}$ of the system (8).

Proof. From Theorem 1 we have that

$$\nu(x) > 0,\ D^+\nu(x) < 0,\ \forall (t,\, x)\ \mathcal{T}\times\mathcal{B}\ . \tag{44}$$

Since $\mathcal{A} \subset \mathcal{B}$, from (41) we have directly that $x_0 \in \mathcal{A}$ implies both the Liapunov stability of $x_e = 0$ and the fact that $\lim_{t\to+\infty} x(t;\, t_0,\, x_0) = 0$ for for all interconnection matrices E. This proves Theorem 5.

The region $\mathcal{A}$ provided by Theorem 5 in (43) is the largest region of connective stability that is available with the Liapunov function $\nu(x)$ in (18) and the constraints specified in (41). Larger stability regions could be obtained by other types of Liapunov functions as discussed by Weissenberger [25]. Furthermore, by using the result of Theorem 1, one can determine the regions of connective exponential stability, which have the property that any solution starting in the region converges to the equilibrium state faster than an exponential.

VII. HIERARCHIC MODELS

On the basis of a computer study of stability in ecosystems, May [1] suggested "... that our model multi-species communities, for given interaction strength and web connectance, will do better if the interactions tend to be arranged in "blocks" - again a feature observed in many models since as noted by Simon [26], the hierarchic structures "... provide the most viable form for any system of even moderate complexity".

In this section, a decomposition-aggregation approach to stability is developed, which can be used to establish structural properties of large hierarchic models of multispecies communities. By grouping species of a large community into a relatively small number of blocks, the community is decomposed into interconnected subcommunities. Stability of each subcommunity is aggregated in a single function and an aggregated model of the community is formed on the basis of the subcommunity stability functions and the nature of the interactions. Stability of the entire community is predicted by analyzing the lower-order aggregate model.

In the decomposition-aggregation analysis, it is required that each subcommunity be stable. This is similar but not equivalent to the requirement made previously that each species is density dependent. In each subcommunity, species may interact arbitrarily. This added possibility of the decomposition-aggregation approach allows for considering trophic level systems with more than one species that are not density dependent (e.g., a number of competing predators). The approach, however, provides the structural information pertinent only to the subcommunities as a whole and not to individual species themselves. Once a species joins a group, it loses its identity, and the approach treats the groups as interacting agents on a higher hierarchial level - that of the aggregate model.

Let us consider a more general mathematical model than that of equation (1),

$$\dot{x} = f(t, x) \tag{45}$$

where $x(t) \in R^n$ is the population vector of the community; the function $f\colon T \times R^n \to R^n$ is defined, continuous and bounded on $T \times R^n$ so that the solutions $x(t; t_0, x_0)$ of (45) exist for all initial conditions $(t_0, x_0) \in T \times R^n$ and $t \in T_0$.

Let us decompose the model (45) into s subsystems described by the equations

$$\dot{x}_i = g_i(t, x_i) + h_i(t, x), \quad i = 1, 2, \ldots, s \tag{46}$$

where $x_i \in R^{n_i}$ is the i-th "vector component" of the population vector

$$x(t) = [x_1^T(t)\ x_2^T(t)\ \ldots\ x_s^T(t)]^T . \tag{47}$$

In (46), $g_i\colon T \times R^{n_i} \to R^{n_i}$ describes the i-th isolated system

$$\dot{x}_i = g_i(t, x_i), \quad i = 1, 2, \ldots, s \tag{48}$$

which is obtained from (45) when all interactions $h_i\colon T \times R^n \to R^{n_i}$ among the subsystems are set to zero $(h_i(t, x) \equiv 0)$. We assume that each $g_i(t, 0) = 0$, $\forall t \in T$, so that $x_{ie} = 0$ is the equilibrium of the model (48).

We assume that the stability properties of each subcommunity with population vector $x_i(t)$ are aggregated by a scalar Liapunov function $v_i\colon T \times R^{n_i} \to R_+$ such that $v_i(t, x_i) \in C^{(1,1)}(T \times R^{n_i})$, and

$$\begin{gathered} \eta_{i1}\|x_i\| \le v_i \le \eta_{i2}\|x_{i2}\| \\ \dot{v}_i \le -\eta_{i3}\|x_{i3}\| \\ \|\text{grad } v_i\| \le \eta_{i4}, \quad \forall (t, x_i) \in T \times R^{n_i} \end{gathered} \tag{49}$$

where $\eta_{i1}, \eta_{i2}, \eta_{i3}, \eta_{i4}$ are positive numbers, and

$$\dot{v}_i = \frac{\partial}{\partial t} v_i + (\operatorname{grad} v_i)^T g_i \ . \tag{50}$$

Properties (49) of the function $v_i(t, x_i)$ imply that each subcommunity when isolated from the rest of the community as described by equation (48), is globally exponentially stable.

We consider the interactions $h_i(t, x)$ among subcommunities to have the form

$$h_i(t, x) \equiv h_i(t, e_{i1}x_1, e_{i2}x_2, \ldots , e_{is}x_s) \tag{51}$$

since we want to apply the connective stability concept to groups of species rather than to the individual species in the community (1). Furthermore, we assume that the interactions among species are bounded and that they satisfy the "connical" constraints

$$||h_i(t, x)|| \leq \sum_{j=1}^{s} e_{ij}\xi_{ij}||x_j|| \ , \ \forall (t, x) \in \mathcal{T} \times \mathcal{R}^n \tag{52}$$

where ξ_{ij} are nonnegative numbers.

Since the estimates (49) of the stability index v_i of subcommunities are given in terms of the Euclidean norm, inequality (52) can be expressed by v_j . This leads directly to formulation of the comparison inequality in terms of the aggregate variables $v_1, v_2, \ldots , v_s$, which can be regarded as components of the vector Liapunov function

$$v = (v_1 \ v_2 \ \ldots \ v_s)^T \tag{53}$$

proposed by Bellman [27] and Matrosov [28] (also Lakshmikantham and Leela [29]).

The stability analysis starts on the subcommunity level, where Liapunov's functions $v_i(t, x)$ are constructed with the estimates (49) for each exponentially stable subcommunity (48). Then, the total time derivative $\dot{\tilde{v}}(t, x_i)$ is taken along the solutions of the corresponding interconnected subcommunity described by (46),

$$\dot{\tilde{v}}_i = \dot{v}_i + (\operatorname{grad} v_i)^T h_i(t, x) \ , \ i = 1, 2, \ldots , s \ . \tag{54}$$

By using the estimates (49) and constraints (52), we can rewrite (54) as

$$\begin{aligned} \dot{\tilde{v}}_i &\leq \dot{v}_i + ||\operatorname{grad} v_i|| \ ||h_i(t, x)|| \\ &\leq -\eta_{i2}^{-1} \eta_{i3} v_i + \sum_{j=1}^{s} e_{ij}\xi_{ij}\eta_{j1}^{-1}\eta_{i4} v_j \\ &\qquad i = 1, 2, \ldots , s \ . \end{aligned} \tag{55}$$

Differential inequalities (55) define the aggregate model of the community,

$$\dot{v} \leq Av \ , \tag{56}$$

where v is the s-vector Liapunov function, and $A = (a_{ij})$ is the $s \times s$ aggregate matrix with elements specified by

$$a_{ij} = \begin{cases} -\eta_{i2}^{-1} \eta_{i3} + e_{ii}(t) \ \xi_{ii} \ \eta_{i1}^{-1} \ \eta_{i4} \\ e_{ij}(t) \ \xi_{ij} \ \eta_{j1}^{-1} \ \eta_{i4} \end{cases} \ . \tag{57}$$

We define as in (15) the matrix $\bar{A} = (\bar{a}_{ij})$ which corresponds to the fundamental interconnection matrix $\bar{E}$ and has elements $e_{ij}(t)$ replaced by $\bar{e}_{ij}$ of $\bar{E}$,

$$\bar{a}_{ij} = \begin{cases} -\eta_{i2}^{-1}\eta_{i3} + \bar{e}_{ij}\,\xi_{ii}\,\eta_{i1}^{-1}\,\eta_{i4} \\ \bar{e}_{ij}\,\xi_{ij}\,\eta_{j1}^{-1}\,\eta_{i4} \end{cases} \tag{58}$$

We assume that $-\eta_{i2}^{-1}\eta_{i3} + \bar{e}_{ii}\,\xi_{ii}\,\eta_{i1}^{-1}\,\eta_{i4} < 0$, $\forall i = 1, 2, \ldots, s$ and is a Metzler matrix.

We recall that $\bar{e}_{ij}$ is one if the j-th subcommunity influences the i-th community, and $\bar{e}_{ij}$ is zero otherwise.

Now, we prove the following:

Theorem 6. The equilibrium $x_e = 0$ of the system (46) is exponentially connectively stable in the large if the corresponding $s \times s$ constant Metzler matrix $\bar{A}$ defined in (58) is a quasidominant diagonal matrix.

Proof. Let us consider a decrescent, positive definite, and radially unbounded function $\nu\colon \mathcal{R}_+^s \to \mathcal{R}_+$,

$$\nu(v) = d^T v \tag{59}$$

where $d > 0$ is a constant s-vector, as a candidate for Liapunov's function for the model (46). Taking the total time derivative $\dot{\nu}$ along the solutions $x(t;\, t_0,\, x_0)$ of (46), we get

$$\dot{\nu}(v) = b^T \dot{v}[x(t;\, t_0,\, x_0)] \quad . \tag{60}$$

Premultiplying inequality (56) by $b^T > 0$, we obtain

$$\dot{\nu} \leq b^T \bar{A} v \ , \tag{61}$$

where the matrix $\bar{A}$ is quasidominant diagonal. We rewrite (61) as

$$\begin{aligned} \dot{\nu} &\leq \sum_{i=1}^{s} d_i \sum_{j=1}^{s} \bar{a}_{ij}\, v_j \\ &\leq \sum_{j=1}^{s} v_j \sum_{i=1}^{s} d_i\, \bar{a}_{ij} \\ &\leq \sum_{j=1}^{s} v_j d_j |\bar{a}_{jj}| + \sum_{j=1}^{s} d_i |\bar{a}_{ij}| \\ &\leq -\pi\nu \ , \qquad \forall t \in \mathcal{T}_0 \ , \ \forall (t_0,\, x_0) \in \mathcal{T} \times \mathcal{R}^n, \ \ \forall E \end{aligned} \tag{62}$$

where $\pi > 0$ is defined by (21).

By integrating the last inequality in (62), we get inequality (23) which in this case can be manipulated to get the exponential estimate (24) on the solution $x(t;\, t_0,\, x_0)$ with $\Pi = s^{1/2} d_M d_m^{-1} \eta_{M2} \eta_{m1}^{-1}$, $d_M = \max_i d_i$, $d_m = \min_i$ $\eta_{M2} = \max_i \eta_{i2}$, $\eta_{m1} = \min_i \eta_{i1}$. This proves Theorem 6.

To illustrate the application of Theorem 6, let us consider a linear community model

$$\dot{x} = Px \ , \tag{63}$$

where P is a time-varying $n \times n$ matrix. The (63) is decomposed into s linear subcommunities describing different trophic levels,

$$\dot{x}_i = P_i x_i \ , \ i = 1, 2, \ldots, s \tag{64}$$

where P are constant $n_i \times n_i$ matrices. The interconnected subcommunities are represented by

$$\dot{x} = P_i x_i + \sum_{j=1}^{s} e_{ij}(t)\, Q_{ij} x_j \ , \ i = 1, 2, \ldots , s \ . \tag{65}$$

In (65), Q_{ij} is an $n_i \times n_j$ constant interconnection matrix, and $e_{ij}(t)$ are elements of the interconnection matrix such that $e_{ii}(t) \equiv 0$, $i = 1, 2, \ldots , s$ (i.e., no feedback in each subcommunity).

We assume nothing about the sign pattern in P_i's since we require only that they are stable. Therefore, there exists a quadratic Liapunov function $V_i(x_i) = x_i^T H_i x_i$, whose total time derivative along the solutions of (64) is $\dot{V}_i(x_i) = -x_i^T G_i x_i$ where $G_i = -(P_i^T H_i + H_i P_i)$ and H_i, G_i are both symmetric and positive definite $n_i \times n_i$ matrices. Therefore, we have the "quadratic" estimates

$$\lambda(H_i)||x_i||^2 \le V_i \le \Lambda(H_i)||x_i||^2$$

$$\dot{V}_i \le -\lambda(G_i)||x_i||^2$$

$$||\text{grad } V_i|| \le 2\Lambda(H_i)||x_i|| \tag{66}$$

where λ and Λ are the minimum and maximum eigenvalues of the indicated matrices. For aggregation purposes, we replace the function V_i by $V_i^{1/2}$ for which we have

$$v_i = V_i^{1/2} \ , \ \dot{v}_i = \frac{1}{2} V_i^{-1/2} \dot{V}_i \ , \ \text{grad } v_i = \frac{1}{2} V_i^{-1/2} \text{ grad } V_i \ . \tag{67}$$

The function $v_i(x_i)$ has the "linear" estimates (49) with

$$\eta_{i1} = \lambda^{1/2}(H_i), \ \eta_{i3} = \frac{1}{2}\Lambda^{-1/2}(H_i)\ \lambda(G_i)$$

$$\eta_{i2} = \Lambda^{1/2}(H_i), \ \eta_{i4} = \Lambda(H_i)\ \lambda^{-1/2}(H_i) \tag{68}$$

which are obtained by using (66) and (67).

To get a bound on interactions among communities $Q_{ij}x_j$, we can use the norm $||Q_{ij}|| = \Lambda(Q_{ij}^T Q_{ij})$.

Now, the total time derivative $\dot{v}_i(x_i)$ along the solutions of the model with interactions present, is

$$\begin{aligned}
\dot{v}_{\sim i} &= \dot{v}_i + (\text{grad } v_i)^T \sum_{j=1}^{s} e_{ij}(t) Q_{ij} x_j \\
&\le \dot{v}_i + ||\text{grad } v_i|| \sum_{j=1}^{s} e_{ij}(t)||Q_{ij}||\ ||x_j|| \\
&\le -\eta_{i3}||x_j|| + \eta_{i4} \sum_{j=1}^{s} e_{ij}||Q_{ij}||\ ||x_j|| \\
&\le -\eta_{i2}^{-1}\eta_{i3} v_i + \eta_{i4} \sum_{j=1}^{s} e_{ij}(t)\Lambda(Q_{ij}^T)\eta_{j1}^{-1} v_j \ .
\end{aligned} \tag{69}$$

Finally, we define the elements of the aggregate matrix $A = (a_{ij})$ as

$$a_{ij} = \begin{cases} -\frac{1}{2}\Lambda^{-1}(H_i)\lambda(G_i) + e_{ii}(t)\Lambda(Q_{ii}^T Q_{ii})\eta_{i1}^{-1}, & i = j \\ e_{ij}(t)\Lambda(Q_{ij}^T Q_{ij})\Lambda(H_i)\lambda^{-1/2}(H_i)\lambda^{-1/2}(H_j), & i \ne j \end{cases} \tag{70}$$

and obtain the aggregate model (56) for the community model (63). Stability of the aggregate model is assured by the quasidiagonal property of the aggregate matrix $\bar{A}$ defined by (70) and $\bar{E}$ as in (58). Stability of the aggregate model is connective and, therefore, structural perturbations among the subcommunities can be tolerated as far as stability is concerned. This result can be applied to the case considered by Paine [10] in which removal of one species from an intertidal community of marine invertebrates led to a severe reduction of the number of species in the community. Our result can establish stability of the reduced community whenever the original community is stable. This is accomplished by considering various numbers of species as subcommunities (subsystems), and considering subcommunities to be disconnected from the community.

Since most of the mathematical models of population dynamics have multiple equilibria, stability is restricted to a finite region of the population space. The decomposition-aggregation stability analysis was extended by Weissenberger [25] to estimation of stability regions using several Liapunov functions.

VIII. CONCLUSION

Within the formal framework of connective stability in the sense of Liapunov, the structural properties of multispecies communities are characterized explicitly, which enabled both qualitative and quantitative study of the effects of increased complexity on community stability. It was shown that in stable communities with density-dependent species, reliability of stability is high, and that it can tolerate a wide range of time-varying perturbations and nonlinearities.

Hierarchial organization of multispecies communities were also studied by the decomposition-aggregation stability method. It was shown that the aggregate community models can possess similar stability properties with respect to blocks of species as the original community matrices with regard to the individual species in the community.

Instability and stability regions of multispecies communities were also considered in the context of connective stability.

There is an apparent connection of general models treated in this work and the broad varieties of specific community models which abound in the field and laboratory studies, and much remains to be done to exploit this connection and arrive at new important results.

IX. REFERENCES

1. May, R. M., "Stability and Complexity in Model Ecosystems", Princeton University Press, Princeton, 1973.

2. MacArthur, R. H., "Species Packing and Competitive Equilibrium for Many Species", Theor. Pop. Biol., 1(1970), 1-11.

3. Gardner, M. R., and W. R. Ashby, "Connectedness of Large Dynamical (Cybernetic) Systems: Critical Values for Stability", Nature, 228(1970), 784.

4. May, R. M., "Will a large complex system be stable?", Nature, 238(1972), 413-414.

5. Šiljak, D. D., "Connective Stability of Complex Ecosystems", Nature, 249 (1974), 280.

6. Šiljak, D. D., "Stability of Large-Scale Systems Under Structural Perturbations", IEEE Trans. Systems, Man, Cybernetics, SMC-2(1972), 657-663.

7. Šiljak, D. D., "On Stability of Large-Scale Systems Under Structural Perturbations", IEEE Trans. Systems, Man, Cybernetics, SMC-3(1973), 415-417.

8. Tanner, J. T., "Effects of Population Density on Growth Rates of Animal Populations", Ecology, 47(1966), 733-745.

9. Harrary, F., R. Norman, and D. Cartwright, "Structural Models: An Introduction to the Theory of Directed Graphs, New York, Wiley, 1965.

10. Paine, R. T., "Food Web Complexity and Species Diversity", Amer. Natur., 100(1966), 65-75.

11. McKenzie, L., "Matrices with Dominant Diagonals and Economic Theory", Proc. Symp. Math. Methods Soc. Sci., Stanford: Stanford University Press, 1966, pp. 47-62.

12. Margalef, R. "Perspectives in Ecology Theory", Chicago: Chicago University Press, 1968.

13. Levins, R., "Ecological Engineering: Theory and Practice, Quart. Rev. Biol., 43(1968), 301-305.

14. Quirk, J., and R. Saposnik, "Introduction to General Equilibrium Theory and Welfare Economics, New York: McGraw-Hill, 1968.

15. Hahn, W., "Stability of Motion", New York, Springer Verlag, 1967.

16. Gantmacher, F. R., "The Theory of Matrices", Vol. II, New York: Chelsea, 1960.

17. Rosenbrock, H. H., "A Liapunov Function for Some Naturally Occuring Linear Homogeneous Time-Dependent Equations", Automatica, 1(1963), 97-109.

18. Sandberg, I. W., "Some Theorems on the Dynamic Response of Nonlinear Transistor Networks", Bell Tech. J., 48(1969), 35-54.

19. Persidskii, S. K., "Investigating the Stability of Solutions of Systems of Differential Equations (in Russian), Prikl. Mat. Mekh., 34(1970), 219-226.

20. Bellman, R., "Topics in Pharmacokinetics - I: Concentration-Dependent Rates", Math. Biosci., 6(1970), 13-17.

21. Šiljak, D. D., "Connective Stability of Nonlinear Matrix Systems", Proc. 1974 IEEE Internatl. Symposium Circuits Systems, San Francisco, 1974. pp. 37-41.

22. Watt, K.E.F., "Ecology and Resource Management", New York: McGraw-Hill, 1968.

23. Grujić, Lj. T., and D. D. Šiljak, "Asymptotic Stability and Instability of Large-Scale Systems", IEEE Trans. Auto. Control, AC-18(1973), 636-645.

24. Ayala, F. J., M. E. Gilpin, and J. G. Ehrenfeld, "Competition Between Species: Theoretical Models and Experimental Tests", Theor. Pop. Biol., 4(1973), 331-356.

25. Weissenberger, S., "Stability Regions of Large-Scale Systems", Automatica, 9(1973), 653-663.

26. Simon, H. A., "The Organization of Complex Systems", in "Hierarchy Theory", H. H. Pattee, Editor, New York: George Braziller, 1973, pp. 1-27.

27. Bellman, R., "Vector Lyapunov Functions", J. SIAM Con., 1(1962), 32-34.

28. Matrosov, V. M., "On the Theory of Stability of Motion (in Russian), Prikl. Mat. Mekh., 26(1962), 992-1002.

29. Lakshmikantham, V., and S. Leela, "Differential and Integral Inequalities", Vol. I and II, Academic Press, New York, 1969.

CONCURRENCY IN GENERALIZED RADIX NON-RESTORING DIVISION

D. E. ATKINS
Systems Engineering Laboratory
Department of Electrical and
Computer Engineering
The University of Michigan
Ann Arbor, Michigan 48104

Ü. KALAYCIOĞLU
Computer, Information and
Control Engineering Program
The University of Michigan
Ann Arbor, Michigan 48104

ABSTRACT

The definition of generalized radix, non-restoring division is reviewed in terms of a recursive relationship and operand range restrictions. The class of divisions considered includes those permitting a redundant quotient digit set and thus quotient selection based upon crude estimates of the divisor and partial remainder. Potentials for concurrency are identified within a set of elementary operations. New techniques which offer complete concurrency between quotient selection and partial remainder formation are emphasized. Speed of operation in such structures, as in multiplication, is limited only by the time to perform multiple operand addition. Means for finding time-optimal structures based upon relative execution time of the elementary operations are discussed. Expected operation times are given for example structures using these techniques.

INTRODUCTION

This paper describes work in progress related to the identification and application of concurrency in the execution of generalized radix, non-restoring division. The class of division algorithms considered include those in which redundancy is introduced into the representation of the quotient thereby permitting quotient digits to be generated (selected) by crude estimates of the divisor and partial remainders. The basis for these techniques is found in [1]. Reference [2] treats the design, cost and performance of a large class of such higher radix, non-restoring schemes. The primary development reported here relates to concurrency between quotient digit generation and formation of the partial remainder.

Non-restoring division is defined by the following recursive relationship and range restrictions:

$$\frac{1}{r} p_{i+1} = p_i - q_i d \text{ for } i = 0, \ldots, m-1 , \tag{1}$$

where

p_0 is the dividend,

p_i is the partial remainder used in the i^{th} recursion,

$(1/r)p_m$ is the remainder,

i is the recursion index,

q_i is the i^{th} quotient digit,

d is the divisor, and

r is the radix (base) of the fixed-base, positional number system used.

Additional definitions and range restrictions are as follows:

(1) The i^{th} quotient digit, q_i, is an element of the set $\{-n, -n+1, \ldots, 0, 1, \ldots, n\}$, where n is an integer such that $n \geq (r-1)/2$.

(2) The dividend, p_0, must be normalized prior to recursion to a range defined by $|p_0| \leq \rho r |d|$, where ρ, the redundancy ratio of the representation of the quotient, is given by $\rho = n/(r-1)$.

(3) The divisor must be normalized to a specified range,

$$a \leq |d| \leq b .$$

(4) Every q_i, for i = 0 through m-1, must be chosen such that equation (1) yields $\frac{1}{r} p_{i+1}$ in the range

$$\left| \frac{1}{r} p_{j+1} \right| \leq \rho |d| .$$

Reference [2] includes additional discussion of these restrictions and definition of another form of the recursive relationship, namely,

$$p_{i+1} = rp_i - q_{i+1} d, \quad i = 0, 1, \ldots, m-1 . \tag{2}$$

In this case $q_0 \equiv 0$. In equation (1) q_i may be non-zero. This property is essential to the "quotient look-ahead" techniques described in this paper.

The design of the quotient selector is realized using the P-D plot techniques studied in [1] and [3]. In a p_i vs d plot (P-D plot), upper and lower boundaries of each quotient digit are defined by straight lines passing through the origin with slopes q_i+e and q_i-e, respectively, where e is defined as the estimate error ratio. It is a function of the amount of estimation in p_i and d. The range of e is $[1/2, \rho]$. For $e > 1/2$, the area covered by two adjacent quotient digits overlap (Figure 1), and the area of the overlap region increases as e increases. Since in an overlap region either quotient digit can be selected, rectangular areas are defined to yield quotient digit values. As e increases, the rectangles can cover greater areas and the quotient digits can be selected on more crude estimates of p_i and d, hence yielding a less costly quotient selector. Therefore, e has been given the value ρ in all previous designs.

ELEMENTARY OPERATIONS

The register-transfer level operations implied in computing one iteration of equation (1) are as follows:

(1) Generate (select) a quotient digit, q_i.
(2) Perform the multiplication, $q_i d$.
(3) Perform the difference operation between p_i and $q_i d$.
(4) Multiply the result of step (3) by r . (A left shift in fixed-base positional number systems.)

These activities are classified into suboperations as shown in Table 1. The description of QS is deliberately vague since eight possible variations are identified in analyzing potential concurrency. The eight variations are defined by all combinations formed by selecting one attribute from each of the properties shown in Table 2.

Activity	Abbreviation	Description
Divisor Multiple Formation	DMF	Given q_i in a specified code, produce $q_i d$ at one input to the adder used for partial remainder formation.
Partial Remainder Formation	PRF	Given stable input to an adder, (p_i and $-q_i d$) form the sum and store result shifted left one digit position (× r).
Quotient Section	QS	Given a partial remainder and divisor find a radix r quotient digit.

Table 1. Description of elementary operations.

Property	Description	Choices
RADIX	Radix of the table lookup division in the quotient selector. ($r = 2^k$, where k is the number of bits of quotient stored per iteration.)	r or r^2
REMAINDER ESTIMATE	In the k^{th} iteration, is an estimate of the next partial remainder, $\hat{p}_{k+1}$ computed in the quotient selector?	Yes , No
REDUNDANCY IN REMAINDER	Is the full precision partial remainder represented in a number system such as produced by a carry-save adder.	Yes , No

Table 2. Definition of QS variations.

Figure 2 names the various alternatives. The emphasis in this paper will be on enumerating the four cases defined at the second level of the tree in Figure 2. The structural differences between the non-redundancy (e. g. , QS1') and redundant (e. g. , QS1") cases are relatively minor compared to differences at the second level.

POTENTIALS FOR CONCURRENCY

Figures 3(a) through 6(a) are examples of concurrency available between the elementary operations defined above. Figures 3(b) through 6(b) are block diagrams of the structure dictated by the corresponding choice of a quotient selector.

QS1

The structure designated QS1 permits no concurrency. At the beginning of the i^{th} iteration, p_i is available. An estimate of p_i, $\hat{p}_i$, is formed (by truncation or truncation and rounding) and applied together with an estimate of the divisor, $\hat{d}$, to QS1. The output, q_i, is used to select the multiple of the divisor to be applied to the adder in the partial remainder formation (PRF) box. After the adder output, p_{i+1}, is stable, it is stored and the index increased by one. Using a QS1 type selector, the time for each division cycle which produces one, radix r quotient digit is

$$\tau_1 = \tau_{QS1} + \tau_{DMF} + \tau_{PRF} \qquad (3)$$

QS2

With quotient selectors of type QS2, QS3, or QS4, some concurrency in the $i = k^{th}$ iteration is possible between formation of p_{i+1} and q_{i+1}. Its operation is defined by the following sequence:

Given $\hat{p}_i^*$, $\hat{d}$, q_i

(1) If $\hat{p}_i^*$ is redundantly represented, convert it to a non-redundant form.

(2) Form $q_i\hat{d}$. This operation could be overlapped with operation (1) if the later is required.

(3) Form $\hat{p}_{i+1} = r(\hat{p}_i - q_i\hat{d})$. (In PREF of Figure 4.)

(4) Form $q_{i+1} = f(\hat{d}, \hat{p}_{i+1})$ by table lookup.

Ideally, the multiple of the divisor, dq_{i+1} is available by the time p_{i+1} is formed. Figure 4(a) illustrates relative timing for actual design using commercially available TTL. Type 2 QS structures yield a division cycle time given by

$$\tau_2 = \max[\tau_{PRF}, (\tau_{QS2} + \tau_{DMF})] \qquad (4)$$

QS3

Quotient selection with a QS Type 3 structure is defined by the following sequence:

Given $\hat{p}_i$, $\hat{d}$, q_i

(1) Same as QS2.

(2) Form $Q_i = f_{r^2}(\hat{d}, \hat{p}_i)$ where f_{r^2} produces a radix r^2 quotient digit by table lookup.

(3) Form $q_{i+1} = Q_i - rq_i$.

Note that computation of $\hat{p}_{i+1}$ is not required. Figure 5(a) illustrates relative timing for an actual design using commercially available TTL. Type 3 QS structures yield a division cycle time given by

$$\tau_3 = \min[\max(\tau_{QS3} + \tau_{DMF}, \tau_{PRF}), \max(\tau_{QS3}, \tau_{DMF} + \tau_{PRF})] \qquad (5)$$

---

* In this case a pseudo estimate, $\hat{p}_i^*$ is formed from a set of bits of p_i such that $\hat{p}_{i+1}$ can be found.

QS4

Quotient selector structure, type 4, is a composite of QS2 and QS3 which offers potential for concurrency between all three elementary operations: PRF, DMF, and QS. Figure 6 illustrates relative timing derived from a paper-design using TTL. Note that successive additions (PRF) are adjacent in time. Type 4 QS structures yield a division cycle time given by

$$\tau_4 = \min[\max(\tau_{QS4}, \tau_{DMF} + \tau_{PREF}, \tau_{PRF}), \max(\tau_{QS4} + \tau_{DMF}, \tau_{PREF}, \tau_{DRF})] \qquad (6)$$

SUPPORTING THEORY

Synthesis and analysis of the proposed division schemes rest upon theory found in [1], [2] and [3] and extensions and refinements which will be reported in [4]. Theory exists to guide the designer in making preliminary cost-performance decisions, and efficient computer programs have been written to support detailed design of quotient selection tables. A cost-performance evaluation between the QS structures proposed in this paper is in progress together with comparison with other high-speed division schemes based on multiplicative rather than additive normalization.

EXAMPLE OF EXPECTED PERFORMANCE

Two design studies have been conducted for a 56 bit by 24 bit divider using commercially available Schottky TTL. The first, based on a type 2 quotient selector in a radix 4 division yields a performance of 32 ns/bit. The times were derived using worst case, not typical delay times. The second, based on a type 4 QS yields a worst case performance of 26 ns/bit.

A non-restoring division using QS2 has also been compared to the Goldschmidt algorithm as described in [5]. Using the same technology and worst case delay times the results are shown in Table 3.

	Non-restoring r = 4 with QS2	Goldschmidt	Ratio
Total Time (worst case)	960 ns	1424 ns	.67
DIP Count	797	2270	.35

Divisor = 24 bits
Dividend = 56 bits
Quotient = 24 bits

Table 3. Comparison of QS2 non-restoring to Goldschmidt division.

ACKNOWLEDGMENTS

The work was supported in part by the National Science Foundation under Grant No. DCR74-18573, a Horace Rackham Faculty Fellowship and Grant, and the Scientific and Technical Research Council of Turkey.

REFERENCES

1. J. E. Robertson, "A new class of digital division methods," IRE Trans. Elec Comp., vol. EC-7, September 1958, pp. 218-222.

2. D. E. Atkins, "An analysis of methods for selection of quotient digits during digital division," (submitted to IEEETC for publication). Available from IEEE Computer Society Repository, No. R74-190.

3. D. E. Atkins, "Higher radix division using estimates of the divisor and partial remainder," IEEE Trans. Comp., vol. C-17, No. 10, October 1968, pp. 925-934.

4. Ü. Kalaycioğlu, "A study of table lookup techniques for quotient digit selection in digital division," Ph. D. thesis in preparation, The University of Michigan, Ann Arbor.

5. S. F. Anderson, et al., "The IBM System/360 Model 91: Floating-point execution unit," IBM Journal of Research and Development, vol. 11, No. 1, January 1967, pp. 34-53.

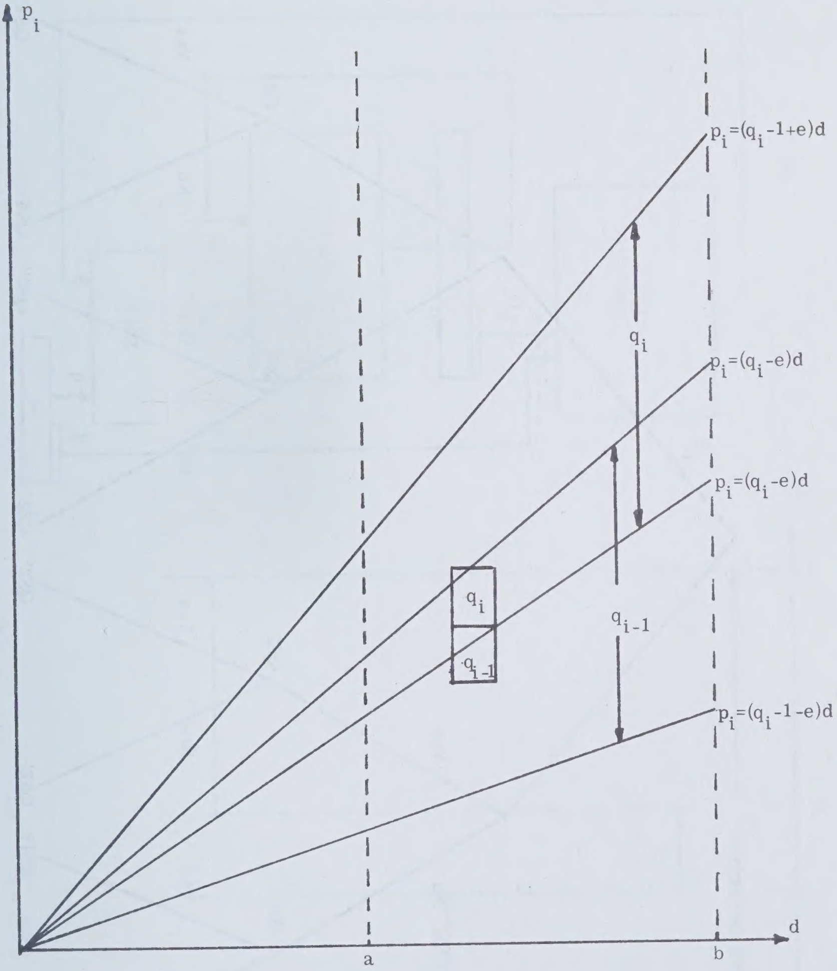

p_i
p_i=(q_i -1+e)d
q_i
p_i=(q_i -e)d
p_i=(q_i -e)d
q_i
q_{i-1}
q_{i-1}
p_i=(q_i -1-e)d
d
a
b

Figure 1. Portion of a P-D plot.

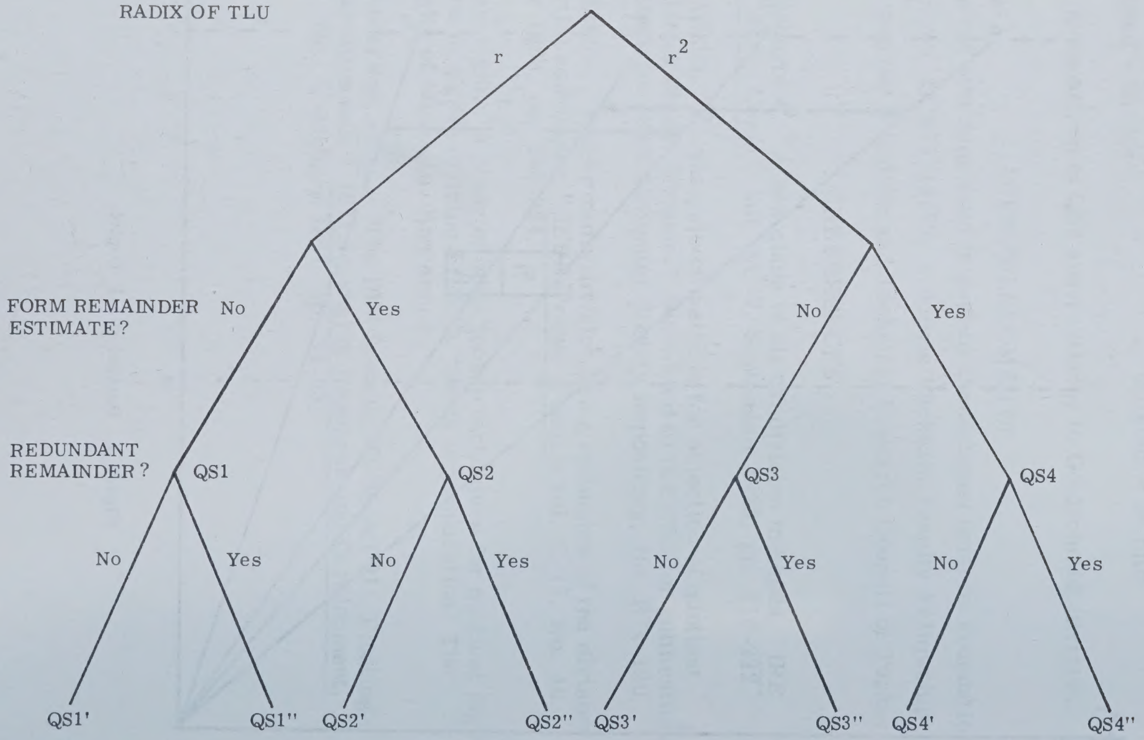

RADIX OF TLU
r
r^2
FORM REMAINDER ESTIMATE?
No
Yes
No
Yes
REDUNDANT REMAINDER?
QS1
QS2
QS3
QS4
No
Yes
No
Yes
No
Yes
No
Yes
QS1'
QS1''
QS2'
QS2''
QS3'
QS3''
QS4'
QS4''

Figure 2. Variations in quotient selector.

$$\tau_1 = \tau_{QS1} + \tau_{DMF} + \tau_{PRF}$$

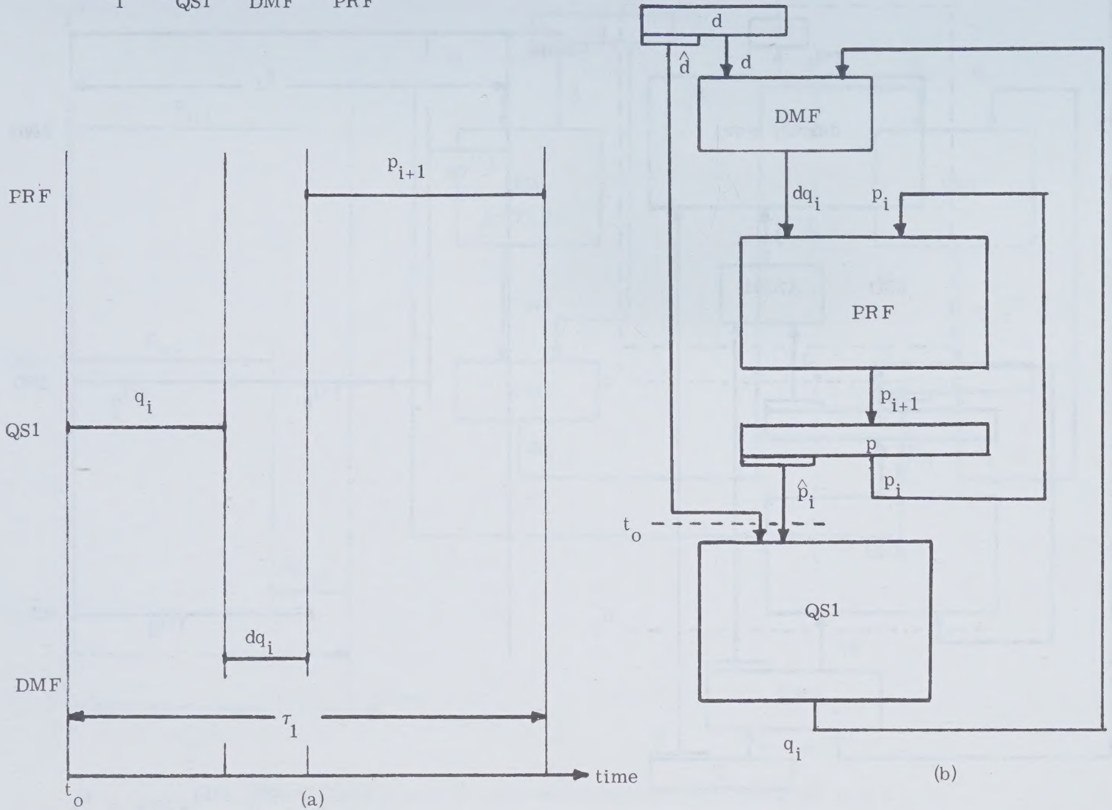

PRF
QS1
DMF
q_i
dq_i
p_{i+1}
τ_1
t_o
time
(a)
d
$\hat{d}$
d
DMF
dq_i
p_i
PRF
p_{i+1}
p
p_i
$\hat{p}_i$
t_o
QS1
q_i
(b)

Figure 3. Example of concurrency and structure using QS, type 1.

$$\tau_2 = \max(\tau_{PRF}, \tau_{QS2} + \tau_{DMF})$$

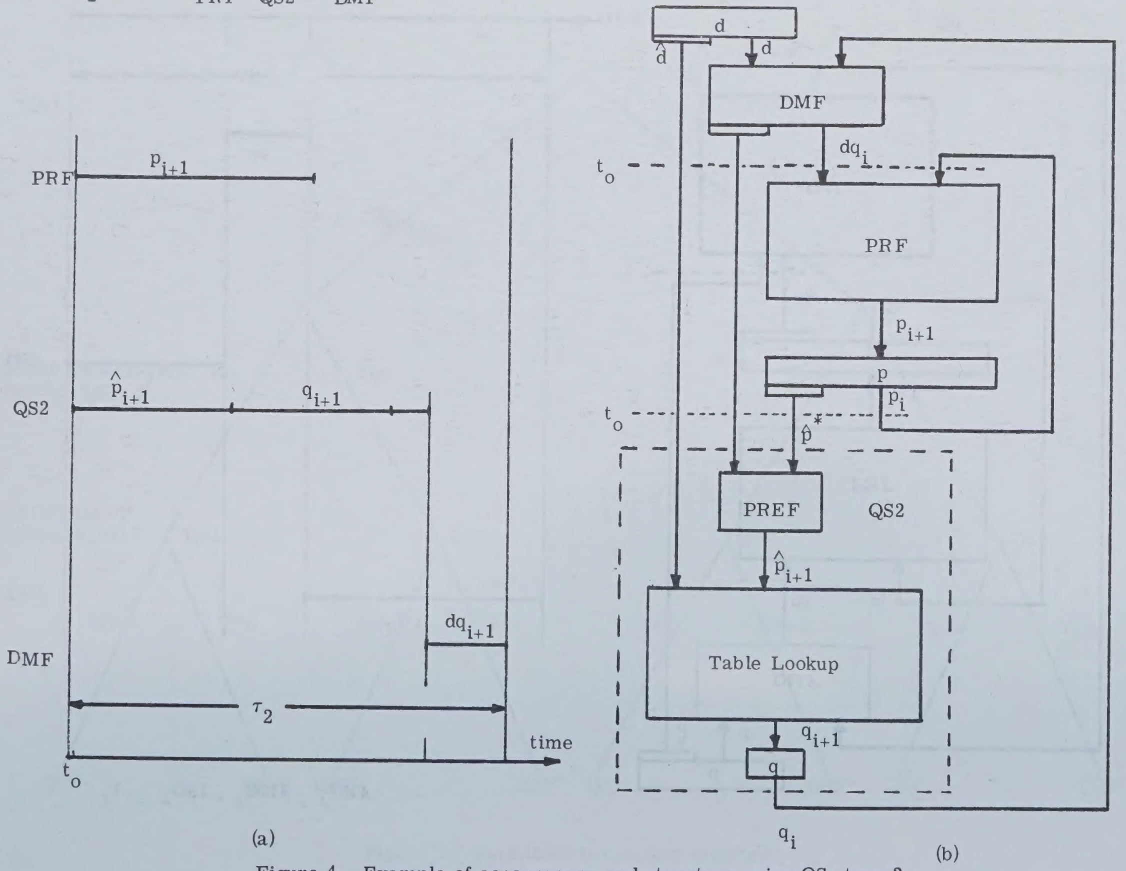

PRF
p_{i+1}
QS2
$\hat{p}_{i+1}$
q_{i+1}
DMF
dq_{i+1}
τ_2
time
t_o
(a)
d
$\hat{d}$
d
DMF
dq_i
t_o
PRF
p_{i+1}
p
p_i
t_o
$\hat{p}^*$
PREF
QS2
$\hat{p}_{i+1}$
Table Lookup
q_{i+1}
q
q_i
(b)

Figure 4. Example of concurrency and structure using QS, type 2.

$$\tau_3 = \min[\max(\tau_{QS3} + \tau_{DMF}, \tau_{PRF}), \max(\tau_{QS3}, \tau_{DMF} + \tau_{PRF})]$$

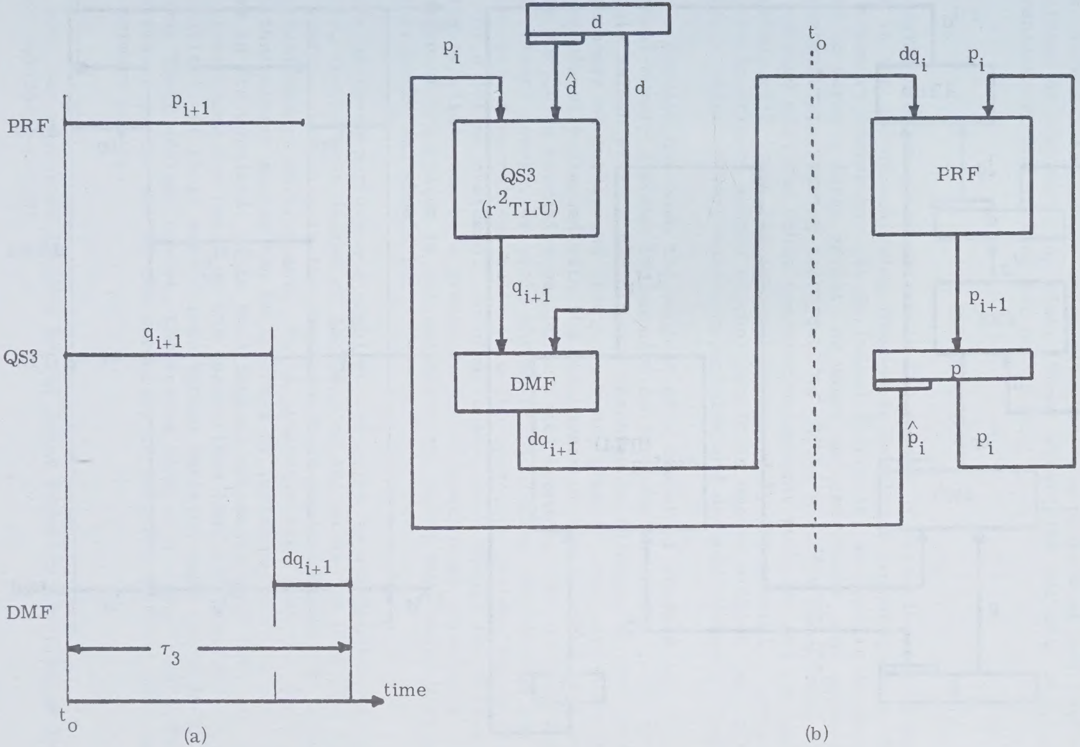

PRF
QS3
DMF
p_{i+1}
q_{i+1}
dq_{i+1}
τ_3
time
t_o
(a)
d
p_i
$\hat{d}$
d
QS3
(r^2TLU)
q_{i+1}
DMF
dq_{i+1}
t_o
dq_i
p_i
PRF
p_{i+1}
p
$\hat{p}_i$
p_i
(b)

Figure 5. Example of concurrency and structure using QS, type 3.

$$\tau_4 = \min[\max(\tau_{QS4}, \tau_{DMF} + \tau_{PREF}, \tau_{PRF}), \max(\tau_{QS4} + \tau_{DMF}, \tau_{PREF}, \tau_{PRF})]$$

QS4

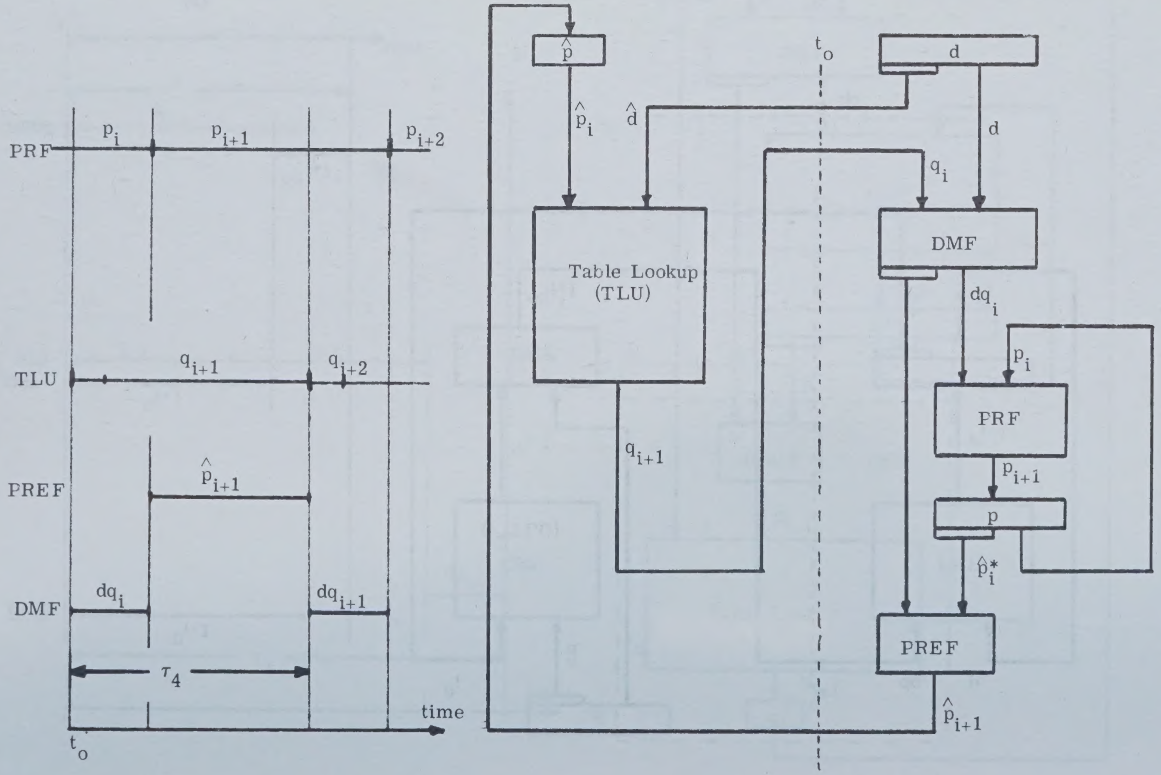

PRF
TLU
PREF
DMF
p_i
p_{i+1}
p_{i+2}
q_{i+1}
q_{i+2}
$\hat{p}_{i+1}$
dq_i
dq_{i+1}
τ_4
time
t_o
$\hat{p}$
$\hat{p}_i$
$\hat{d}$
Table Lookup (TLU)
q_{i+1}
t_o
d
d
q_i
DMF
dq_i
p_i
PRF
p_{i+1}
p
$\hat{p}_i^*$
PREF
$\hat{p}_{i+1}$

Figure 6. Example of concurrency and structure using QS, type 4.

CRITICAL-SET ALGORITHMS FOR PROGRAM LOCALITY IMPROVEMENT[†]

DOMENICO FERRARI
Computer Science Division, Department of Electrical Engineering and
Computer Sciences and the Electronics Research Laboratory
University of California, Berkeley, California

ABSTRACT

A class of algorithms for restructuring programs to improve their performance in a virtual-memory system is presented. The algorithms are strategy-oriented, i.e., they are based on the knowledge of the memory management strategy used by the system on which the programs have to run. Critical-set algorithms reduce the page fault rate of programs running in systems in which the memory space allotted to a program at any instant can be predicted at restructuring time. These strategies include, among others, local replacement and working set policies.

1. INTRODUCTION

A major problem which arises in the design of virtual-memory computer systems is the one of achieving reasonable levels of performance in the system's memory hierarchy. It is indeed difficult, especially in systems designed to serve a large variety of users and user needs, and to do so using the multiprogramming technique, to choose a memory management strategy which, coupled with the other resource management policies and for any type of workload, will cause the information needed by the central processing unit to be in primary memory at the time it is needed without requiring exceedingly large primary-memory capacities and secondary-memory bandwidths.

This problem has been the subject of a number of studies and many of the virtual-memory systems implemented so far have succeeded in attaining acceptable values of such performance measures as throughput, CPU utilization, response or turnaround time. However, some of them have been and are still occasionally plagued with the phenomenon of thrashing, a performance collapse due to an excessive paging activity stemming from an overcommitment of primary memory, or avoid thrashing by a quite complicated and expensive set of mechanisms built into the operating system, which are the source for a lot of overhead. In other cases, even though the risk of thrashing is eliminated by a proper choice of management strategies, the performance of the system is not satisfactory, being too low with respect to what it could and should be.

Most of these performance problems can often be attributed, at least partially, to the poor degree of locality exhibited by the programs running on the system. Locality is the most important aspect of program behavior in a virtual-memory environment. It is generally true that the performance of a virtual-memory system can be improved by improving the locality of the programs in its workload. This is a logical consequence of the fact that memory hierarchies are based on the assumptions that all programs exhibit some locality: if their memory referencing patterns were randomly distributed over their address spaces, the access time of a memory hierarchy would be much closer to the one of the secondary or tertiary devices than to the one of primary memory.

[†] This work was supported in part by the Joint Services Electronics Program Contract F44620-71-C-0087.

Thus, besides investigating new and better memory management strategies, it is fruitful to study methods for improving the locality of programs. Evidently, improving the locality of a program is not only beneficial for the performance of the system but is also profitable for the user, since a more local program is bound to require less system resources and will therefore run faster and less expensively.

A solution to this problem, which does not require any special skills or training on the part of the programmer, is the one called program restructuring [1]. It consists of rearranging parts (called blocks) of an existing program in its virtual address space so as to increase the probability that, when the program is running, the information required to be in primary memory will be found there at the right time, and relatively few interruptions will be necessary.

This paper describes a class of restructuring algorithms which can be used to determine how the blocks of a program should be rearranged. The algorithms are illustrated in Section 3, after having characterized the problem in Section 2. Some examples of the application of the algorithms are discussed in Section 4.

2. RESTRUCTURING IN A PREDICTABLE-SPACE ENVIRONMENT

A restructuring procedure which is becoming very popular is the one which bases the program's rearrangement on information about the program's dynamic behavior gathered by properly instrumenting and running the program before restructuring it. Although the forms in which the information is gathered may differ, they can all be conceptually reduced to the recording of the block reference string generated by the program during execution. This is the string of the references, chronologically ordered, to the blocks into which the program and its data have been previously subdivided.

To simplify our discussion, we shall assume that virtual memory is implemented by the paging mechanism and that the demand fetch policy is used. Blocks will have to be substantially smaller than pages (a rule of thumb [1] requires their average size to be between one tenth and one third of the page size) in order for restructuring to be beneficial.

The restructuring procedure being considered here then applies a restructuring algorithm to the block reference string (or to an equivalent characterization of dynamic behavior) in order to obtain the so-called restructuring graph. This is a non-directed graph whose nodes represent the blocks of the program and whose edges have labels which quantify the "desirability" of grouping into the same page the two blocks defining that edge. Nodes have weights corresponding to the sizes of the blocks they represent. Once the restructuring graph (or the equivalent restructuring matrix) has been constructed, a clustering algorithm is applied to it, which results in a list of recommended groupings of blocks into pages. The clustering algorithm tries to maximize the sum of the labels of edges connecting blocks belonging to the same page, under the constraint that the sum of the node weights within each page be not greater than the page size. Finally, the program is restructured according to the recommendations.

A useful classification of the several known restructuring algorithms which can be used in a procedure of the type described above is the one based on the distinction between strategy-independent and strategy-oriented algorithms. The first category contains those algorithms which construct the restructuring graph without taking into account the memory management strategy under which the program will run, and includes the nearness algorithm [1] and the one by Masuda et al. [2]. The converse is true for strategy-oriented algorithms, which try to combine the knowledge of the program's dynamic behavior, as described by the block reference string, with the one of the memory management strategy in order to improve the results of the restructuring procedure. The second category includes the algorithms described in [3], [4], [5], and those to be discussed in this paper.

An important distinction within the category of strategy-oriented algorithms has to do with the type of strategy being considered: if the strategy is such that the amount of primary memory space to be given to the program is known in advance (either because it is fixed and constant, or because it only depends on the program's behavior and not on the whole workload), then we have a predictable-space strategy; otherwise, we have an unpredictable-space strategy, which is much more difficult to deal with in the context of strategy-oriented restructuring [4]. In this paper, we restrict ourselves to considering predictable-space strategies only, which include local replacement with fixed memory partitioning and working-set strategies (which partition memory dynamically).

What system and/or program performance measure should a restructuring algorithm seek to optimize? This fundamental question has not received, so far, a generally satisfactory answer, essentially because of the difficulty of relating the behavior of a single program to global system performance. Improving the degree of locality is not a satisfactory answer since there is no convincing quantification of it which is independent of the management strategy or of other aspects of the system's organization. A popular measure in memory hierarchy studies is the total number of page faults generated by the program or its page fault rate. Obviously, using this measure requires that the primary memory space in which the program is run be specified at any instant of its execution. Since this is also the assumption on which consideration of predictable-space strategies is based, we shall take the number of page faults as our figure of merit (to be minimized), even though we recognize that, depending on the particular system at hand, other measures, for instance the mean working set size, might be more important and should therefore be considered instead of the one we have chosen.

3. CRITICAL-SET ALGORITHMS

If, as has been assumed in the previous section, the system on which a program to be restructured is going to run uses a predictable-space strategy for managing its memory hierarchy, the knowledge of the strategy can be exploited to determine from the block reference string what blocks would certainly be in primary memory at any given instant if the program with the same input data as in the instrumented run were executed on the system under consideration. To obtain this result, it is sufficient to simulate the running of the program under that strategy dealing with blocks as we would deal with pages, i.e., pretending that the block reference string is the page reference string. The blocks which would certainly be in primary memory at time t under the described circumstances will be called the present blocks at t. Thus, for example, if the strategy is fixed partitioning with local LRU replacement and the program is going to run in a partition composed of 20 page frames, the 20 most recently referenced blocks at time t will be the present blocks at that time.

Of course, since blocks are smaller than pages, there will be many non-present blocks in primary memory at any given time, among which those grouped with the present blocks by the restructuring procedure. However, the identity of these blocks cannot be foreseen before the restructuring is completed, and only a probability of being in memory at a certain instant could be assigned to them.

Let $B = b_1 b_2 \dots b_r$ be the block reference string generated by the program to be restructured and $t_1, t_2, \dots t_r$ the (virtual) time instants at which the block references $b_1, b_2, \dots b_r$ were issued. Let $S(t_k)$ be the set of the present blocks at time t_k, with k integer and $1 \leq k \leq r$. $S(t_k)$ is said to be a critical set if $b_{k+1} \notin S(t_k)$, i.e., if the block referenced at time t_{k+1} is not included into the set of present blocks at time t_k. If $S(t_k)$ is a critical set, b_{k+1} is its critical reference.

The importance of critical sets in the page fault generation process can be seen by transferring the above definitions from the context of blocks to the context of pages. The present pages at t_k are the ones (and the only ones) which are in primary memory at t_k. A critical set of pages is a set which does not contain the page referenced next, and hence a critical reference corresponds to a page fault. It is easy to see that a non-critical reference in the block reference string can never, under any restructuring arrangement, become a critical reference in the page reference string: if b_{k+1} is found to be present in primary memory at t_k, it will always be there at t_{k+1} when it is referenced. Thus, page faults only correspond to critical block references, and when blocks are clustered into pages, some critical block references remain critical (thereby becoming page faults), while others become non-critical, due to the fact that they have been grouped with at least one block which either is present or happens to be in primary memory anyway at that time.

The above discussion suggests that, to minimize the number of page faults, we should maximize the number of critical block references which become non-critical when blocks are clustered into pages. This is the basic objective of critical-set algorithms, which compute the labels of the edges in the restructuring graph so as to increase the probability that blocks which are critical references be grouped with at least one of the members of the corresponding critical set, so that, by virtue of being in the same page as a present block, they can be found in primary memory when they are requested by the CPU.

The simplest way to implement this principle is perhaps the one of incrementing by 1 the labels of the edges connecting each critical-reference node to the present-block nodes, that is, if the critical set is $S(t_k) = \{b_{k1}, b_{k2}, \dots b_{km}\}$, of incrementing by 1 the labels of (b_{k+1}, b_{k1}), (b_{k+1}, b_{k2}), $\dots$ (b_{k+1}, b_{km}).

A restructuring algorithm may be defined to be optimum if it constructs a restructuring graph which, clustered by an optimum clustering algorithm, yields groupings of blocks which minimize the number of page faults. That critical-set algorithms are non-optimum is evident: the grouping of two blocks together has a number of effects (e.g., it makes non-critical certain critical sets) which cannot be easily derived from (or represented in) the restructuring graph, and these effects cause some distortions which in general result in a non-optimum solution. The only partial exception is the critical working set algorithm, which applies to systems with a working set strategy (see [3] and [5]): it can be proved [5] that this algorithm is optimum when no more than two blocks are grouped into each page. The exception is only partial since, when there is at least one page with more than two blocks, the proof is not valid anymore and examples in which the algorithm is not optimum can be easily found.

However, critical-set algorithms are better than strategy-independent algorithms: this statement's validity has been experimentally confirmed for the critical working set algorithm [4] [5] but can reasonably be expected to hold for the whole class, due to the facts that they exploit a larger amount of relevant information and that strategy-independent algorithms have been found to suggest rearrangements generally far from the optimum (this has been deduced from experimental evidence that their solutions could be substantially improved by using simple, non-optimum algorithms). Taking the strategy into account makes critical-set (and in general strategy-oriented) algorithms more expensive to implement than strategy-independent ones, such as the nearness algorithm. However, their use has been seen to be still cheap enough as to make them practically applicable and their costs easy to compensate for if programs which are executed many times are restructured.

4. EXAMPLES

The block reference string in Table I will be used as the characterization of the dynamic behavior of a hypothetical program to be restructured. In this highly idealized and highly simplified example, block reference times are assumed to be equally spaced, blocks are supposed to have all the same size, and the block size to be half of the page size.

Four critical-set algorithms have been applied to the block reference string in Table I: they correspond, respectively, to local LRU replacement with s = 3 page frames allotted to the program, local LFU (Least Frequently Used) replacement with s = 3 page frames, local FIFO (First In First Out) replacement with s = 3 page frames and WS (Working Set) with window size w = 3 references. The restructuring matrices resulting from the application of these restructuring algorithms are displayed in Tables II, III, IV and V. The underlined entries in these matrices correspond to the pairs of blocks which are grouped together by the clustering algorithm with the objective of maximizing the sum of these entries. These pairs are listed in Table VI, where they are given arbitrarily the page names a, b, c, d. These names are then used to derive the various page reference strings from the block reference string, as shown in Table I. By simulating the processing of the page reference strings in Table I under the respective strategies which have inspired the various rearrangements of the program, we can compute the numbers of page faults generated by the restructured versions of the program and compare them with those produced by the non-restructured (NR) program. These comparisons are reported in Table VII and show that substantial improvements are obtained by restructuring in all cases. Our examples also show that critical-set algorithms are not optimum: for instance, the solution found for LRU replacement turns out to produce only 7 page faults when run under LFU with s = 3 page frames, whereas the groupings suggested by the LFU algorithm generate 9 page faults.

5. CONCLUSION

A class of strategy-oriented restructuring algorithms has been presented. The algorithms in the class are more expensive to use than strategy-independent ones, but their one-time costs can still be easily compensated for by the performance improvements they produce in programs to be executed sufficiently many times. Critical-set algorithms appear to be promising and deserve more study. In particular, their performance in unpredictable-space environments and their sensitivity to modifications in the memory management strategy should be investigated.

REFERENCES

[1] D.J. Hatfield and J. Gerald, Program restructuring for virtual memory, IBM Systems Journal, vol. 10, no. 3, 1971, 168-192.

[2] T. Masuda, H. Shiota, K. Noguchi and T. Ohki, Optimization of program organization by cluster analysis, Information Processing 74, North Holland, Amsterdam, 1974, 261-265.

[3] D. Ferrari, A tool for automatic program restructuring, Proc. ACM 73, Atlanta, GA., August 1973, 228-231.

[4] D. Ferrari, Improving program locality by strategy-oriented restructuring, Information Processing 74, North Holland, Amsterdam, 1974, 266-270.

[5] D. Ferrari, Improving locality by critical working sets, Communications of the ACM, vol. 17, no. 11, November 1974.

Table I

Sample block reference string and corresponding page reference strings

	1717135626625648848348353535626225611713526262 4883
NR	adadabccaccaccbddbdbbdbcbcbccacaaccaadabcacacabddb
LRU	acacabcbabbacbddddbddbcbcbcbabaacbaacabcababadddb
LFU	aaaaabbccccbcddddbddbbbbbccccbcaaaabbcccccdddb
FIFO	aaaaabcdbddbcdcddcdbcdbcbcbcdbdbbcdaaaabcbdbdbcddb
WS	aaaaabccbccbccddddbddbcbcbccbcbbccaaaabcbcbcbdddb

Table II

Restructuring matrix for LRU replacement with s = 3 pages

	2	3	4	5	6	7	8
1	$\underline{2}$	2	0	3	2	2	0
2		4	2	3	3	0	1
3			2	3	$\underline{5}$	2	2
4				3	2	0	$\underline{2}$
5					3	$\underline{3}$	2
6						1	3
7							0

Table III

Restructuring matrix for LFU replacement with s = 3 pages

	2	3	4	5	6	7	8
1	3	5	3	5	4	$\underline{3}$	3
2		2	1	5	$\underline{7}$	2	0
3			1	$\underline{8}$	6	1	4
4				2	4	0	$\underline{5}$
5					7	2	3
6						3	4
7							0

Table IV

Restructuring matrix for FIFO replacement with s = 3 pages

	2	3	4	5	6	7	8
1	1	2	0	3	1	$\underline{2}$	0
2		$\underline{5}$	2	3	3	2	2
3			2	3	4	2	2
4				$\underline{3}$	2	0	2
5					3	2	1
6						2	$\underline{3}$
7							0

Table V

Restructuring matrix for a working set strategy with w = 3 references

$$\begin{array}{c|ccccccc}
 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
\hline
1 & 2 & 2 & 0 & 3 & 2 & \underline{2} & 0 \\
2 & & \underline{3} & 2 & 5 & 4 & 0 & 1 \\
3 & & & 2 & 3 & 3 & 2 & 2 \\
4 & & & & 2 & 2 & 0 & \underline{2} \\
5 & & & & & \underline{6} & 2 & 2 \\
6 & & & & & & 1 & 2 \\
7 & & & & & & & 0
\end{array}$$

Table VI

Groupings of blocks recommended by critical-set algorithms

Page	NR	LRU	LFU	FIFO	WS
a	1,2	1,2	1,7	1,7	1,7
b	3,4	3,6	3,5	2,3	2,3
c	5,6	5,7	2,6	4,5	5,6
d	7,8	4,8	4,8	6,8	4,8

Table VII

Performances of the restructuring algorithms

Memory management strategy	Number of page faults	
	Non-restructured program	Restructured program
LRU (s = 3 pages)	10	6
LFU (s = 3 pages)	16	9
FIFO (s = 3 pages)	15	8
WS (w = 3 references)	14	11

A MODULAR SISD PROCESSOR

R. G. CORNELL
Bell Laboratories
Naperville, Illinois

ABSTRACT

A modular, single instruction stream-single data stream (SISD) processor based on a strict hierarchy of component microprocessors is presented. The processor architecture is endowed with variable (or growable) operational capacity based upon the number of component microprocessors employed. This type of processor is seen to be most relevant to special purpose and control applications, where the stored program is fixed.

1. INTRODUCTION

Although the true "computer on a chip" has not yet been realized, the typical LSI microprocessor (μprocessor) may be combined with a handful of ancillary devices to form computers exhibiting "small scale" processing power. Progressing a step further, we address ourselves herein toward the use of the μprocessor as a building block in the construction of computers of "medium scale" processing capacity. It has been observed[1] that man's computational perception has been influenced through the inherently sequential nature of his mathematics and 15 years of FORTRAN programming. Consequently, we find it currently advantageous to consider machines of the single instruction stream, single data stream (SISD[2]) variety in order to exploit the intuitively appealing sequential programming languages which typically accompany the SISD architecture.

The processor architecture proposed herein utilizes its component μprocessors as distinct processing entities, rather than simply as alternatives to random logic that make up that processor. A simple approach toward "inter-μprocessor" control and coordination is presented, whereby the processor derives its control structure from the structure of its applications program. It follows that this architecture is to be applied to those special purpose and control applications characterized by a fixed (i.e., static) stored program.

2. HIERARCHICAL ORGANIZATION

The multiple μprocessor structure under consideration is as depicted in Figure 1. μprocessor P_1 is directly in communication with its own memory module(s) M_1, and with μprocessor P_2. However, P_1 may access P_2's memory, as indicated by the dashed line which connects the P_1 and P_2

memory buses. In general, P_i sits on the P_{i-1} memory bus and appears to P_{i-1} to be a memory controller for M_i. M_i may be accessed by P_{i-1}, through the unseen intervention and control of P_i. Overhead due to supervisory control and coordination between P_i and P_{i-1} is minimized through a partition of the applications program such that P_i and P_{i-1} operate asynchronously with a minimum of interaction. All normal intercommunication between P_i and P_{i-1} is conducted via messages left in a buffer area of M_i. This mode of intercommunication is hierarchical in nature, where P_i functions as a secondary processor to P_{i-1}.

The hierarchical structure defined is readily extended to a system of n μprocessors. P_i's address space includes all memory "downstream" (i.e., M_j, $j \geq i$), and none of the memory "upstream" (Figure 2). A μprocessor is aware of (i.e., monitors the memory requests of) the next processor upstream, but does not see any μprocessors downstream. Finally, each μprocessor is endowed with a private I/O port, and thus has direct access to peripheral units not normally connected to the other μprocessors. In such an organization, the first μprocessor in the chain (P_1) serves as the "principle" μprocessor, for it alone has access to the entire memory. Similar to a bureaucratic hierarchy, each successive μprocessor has less supervisory responsibility than its superior, and therefore should have more real time available for task execution. The availability of private I/O ports to each μprocessor reduces the overhead induced by inter-μprocessor requests for I/O information.

3. COMPONENT PROCESSOR ORGANIZATION

The structure of the component μprocessor is that of the classical von Neumann processor, endowed with two independent memory ports (Figure 3). Here, data intercommunication with memory "downstream" is buffered through the memory data register (MDR), with addresses provided via the memory address register (MAR). Meanwhile, the μprocessor monitors each memory request issued by the μprocessor immediately "upstream." Any P_{i-1} memory address request is recorded in the upstream data buffer (UDB) and the upstream address buffer (UAB). P_i then examines the P_{i-1} memory request and determines whether or not the address in UAB is in its own address space, M_j, $j \geq i$. If so, the UAB content is transferred to MAR, (the content of UDB is transferred to MDR in the case of a P_{i-1} "write memory" instruction) and the proper control signaling is issued. For "read memory" operations, the fetched data is passed upstream until it reaches the requesting μprocessor.

It is clear that the execution of any task assigned to P_i should not depend upon any data or program sequences stored exclusively in M_k, $k<i$.

4. PRIORITY ORDERING

The eventual partition of an application program to be executed on this hierarchical system must match the priority ordering imposed upon the component μprocessors. Care must be taken to build the hierarchy of command such that the number of memory requests from P_i to M_j's (where $i<j$) is minimized. This is because each P_i, upon recognition of a memory request to storage in M_k, $k \geq i$, must suspend execution of its current task in order to service the memory request. Also, the amount of overhead during a memory request from P_i to a location in M_j increases as a function of $j-i$.

Two priority orderings are possible, low-to-high or high-to-low with respect to subscripts. In the former case, processor P_i's memory request must be immediately responded to by processor P_{i+1} in interrupt fashion. This represents an attempt to minimize processor P_i's idle time (especially evident when P_i initiates a "read" request). The high-to-low priority ordering observes the convention that P_i need not honor P_{i-1}'s memory request immediately, but may wait until it reaches a convenient state. In this "polling" mode the P_i interrupt overhead is avoided, while P_{i-1}'s waiting time is maximized. The selection of the particular priority ordering to be employed must show consistency with the expected frequency and nature of inter-μprocessor memory requests of the particular applications program.

5. AN APPLICATION FOR A HIERARCHICAL PROCESSOR

The hierarchical organization is only applicable where the program can be partitioned to match the processor-memory relationships defined in hardware. In effect, the program must be structured as exemplified by Figure 4. The main program is partitioned into tasks, some of which actually are imbedded in other tasks. Each task of the same level (as denoted by the number of subscript variables) operates over a distinct data base. The order of execution of tasks of the same level is variable. Tasks which in turn have imbedded tasks cannot complete until all imbedded tasks have completed. Clearly, the example program could be written to be executed in a purely sequential manner, using a programming language typical of those employed with SISD machines (Figure 5A). Alternatively, we might choose to introduce some parallelism by "off-loading" certain tasks to a secondary μprocessor as shown in Figure 5B. Here, no hierarchical structure has emerged in the code as the tasks are presumed independent. Figure 5C shows a further off-loading where independent subtasks are executed in parallel on several μprocessors.

Each μprocessor is controlled by a task dispenser, which monitors an inter-μprocessor buffer for work requests. A request found in the buffer may cause the task dispenser to initiate a particular task program. The task program may execute autonomously within processor P_i, or it may require the help of a processor P_j, $j>i$, in order to meet real-time constraints. If this is the case, a message is left in the M_j inter-μprocessor buffer. P_i periodically interrogates the inter-μprocessor buffer for the results of P_j's computation. At the conclusion of P_i task execution, a completion message is left in the M_i inter-μprocessor buffer and control is returned to the task dispenser.

A nested subprocess program may be generated by examining an application program for independent subroutines, and then off-loading the principal processor by removing those tasks to the next processor in the hierarchy. This process is continued as necessary to load-balance the system. As an example, consider a simplified generic program designed to control telephone switching networks of various sizes (see Figure 6). The program consists of an endless loop, with the three main functions of network scanning, processing service requests, and performing maintenance operations. Each main function is implemented with one or more subloops, each of which may be implemented by subloops. Implementation of this program in sequential fashion on a single μprocessor (P_1) presents no programming problems, but real-time performance might be impossible to achieve for some network sizes. A separate μprocessor (P_2) might be assigned to the processing of service requests. Requests detected by the scanning program are reported to the M_2 inter-μprocessor buffer. P_1 is immediately free to perform maintenance operations and subsequently scan for requests. P_2 may or may not be able to meet real-time demands. If it cannot, a third μprocessor, P_3, could be incorporated to perform call setup only, thus, P_3 would search for an available network path and send the appropriate control signals to the switching network to connect the call. Subsequently, P_2 would interrogate the M_3 inter-μprocessor buffer in order to be aware of which path to tear down in the event of a disconnect request initiated by the P_1 scanning program. The maintenance program executed by P_1 would periodically gain access to all memory. Simulation has shown that, for this example, inter-μprocessor requests do not dominate the real-time of any μprocessor, and few inter-μprocessor requests are made between μprocessors which are not directly connected.

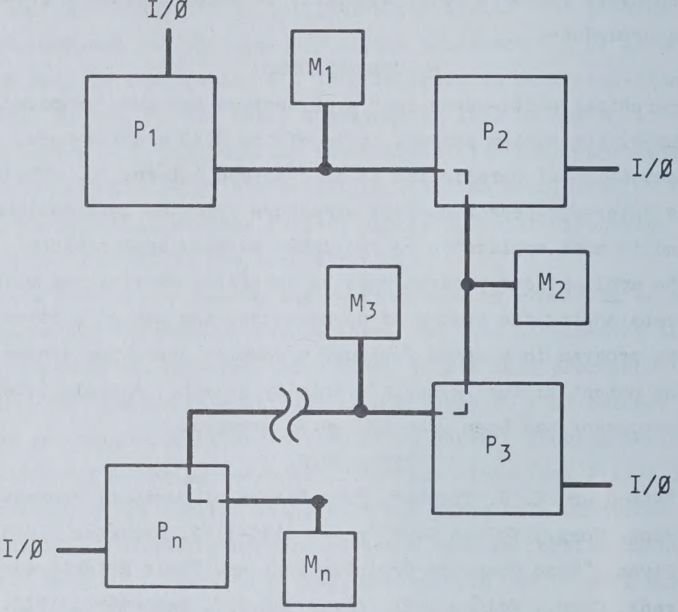
The result is a family of telephone switching processors designed around a single "generic" program, suited for the requirements of many office sizes. To grow a system, additional μprocessors could be added, and program segments subsequently off-loaded to those μprocessors. In addition, software features could similarly be added through a distributed off-loading procedure.

6. CONCLUSIONS

A hierarchical multi-μprocessor architecture has been proposed. To the programmer, the system appears to be of the SISD architecture, although computational parallelism is in evidence internally. The system derives its inter-μprocessor control structure from the applications program, and is most applicable to dedicated machine applications. Although the applications program must be carefully partitioned with a view toward load-balancing the system of μprocessors, the use of a structured applications program in a sense replaces a complex operating system and provides the potential for "generic" capacity growth. A small telephone switching processor has been proposed as an example.

REFERENCES

1. P. C. Patton and K. J. Thurber, "The Future of Parallel Processing," IEEE Trans. Comp., Volume C-22, pages 1140-1142, December, 1973.
2. M. J. Flynn, "Some Computer Organizations and Their Effectiveness," IEEE Trans. Comp., Volume C-21, pages 948-960, September, 1972.


I/0
M1
P1
P2
I/0
M2
M3
P3
I/0
Pn
I/0
Mn

HIERARCHICAL STRUCTURE

FIGURE 1

$P_1 \longrightarrow M_1, M_2, M_3, \ldots\ldots, M_n$

$P_2 \longrightarrow M_2, M_3, \ldots\ldots, M_n$

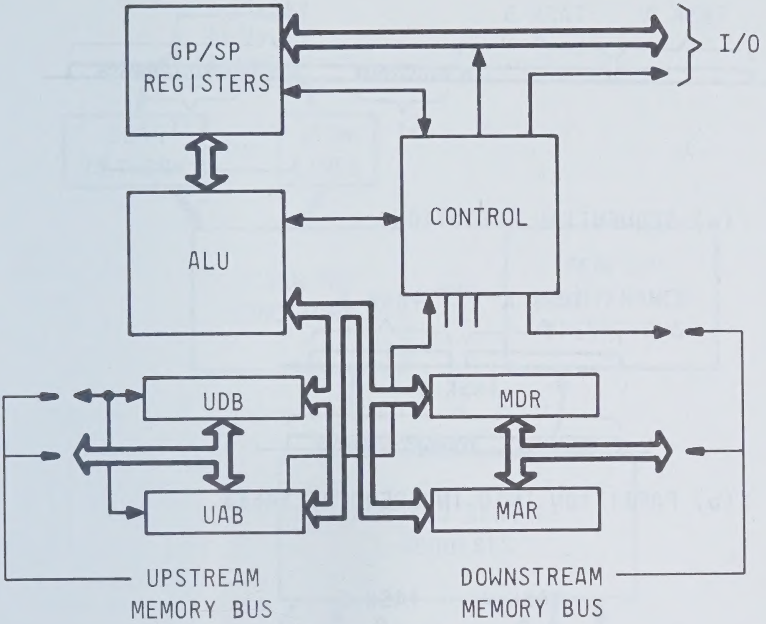
$P_3 \longrightarrow M_3, \ldots\ldots, M_n$

$\vdots$

$P_n \longrightarrow M_n$

PROCESSOR P_i ADDRESS SPACE

FIGURE 2


GP/SP
REGISTERS
I/O
CONTROL
ALU
UDB
MDR
UAB
MAR
UPSTREAM
MEMORY BUS
DOWNSTREAM
MEMORY BUS

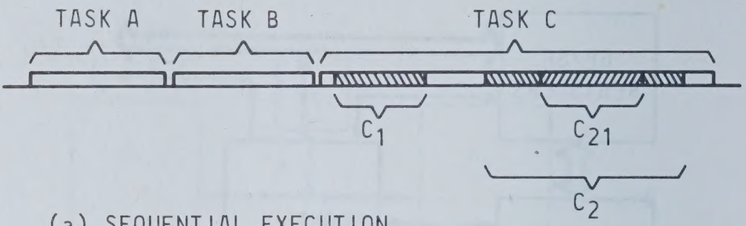
COMPONENT PROCESSOR

FIGURE 3

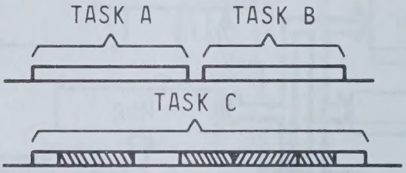
```
BEGIN
    BEGIN  } TASK A
    END
    BEGIN  } TASK B
    END
    BEGIN
        BEGIN  } TASK $C_1$
        END
        BEGIN                                  } TASK C
            BEGIN  } TASK $C_{21}$  } TASK $C_2$
            END
        END
    END
```

PROGRAM STRUCTURE

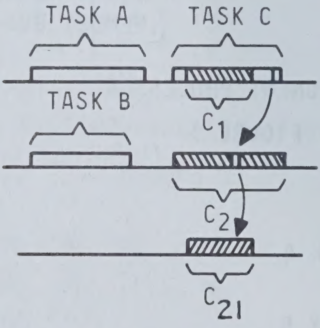
FIGURE 4


TASK A
TASK B
TASK C
C_1
C_{21}
C_2

(a) SEQUENTIAL EXECUTION


TASK A
TASK B
TASK C

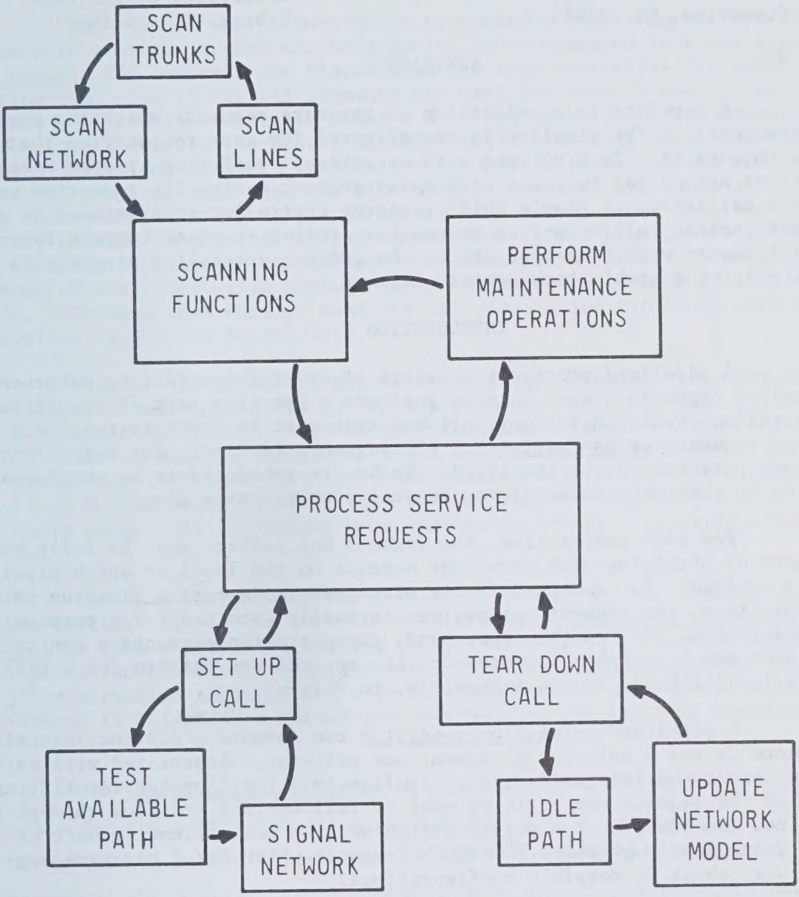
(b) PARTITION INTO INDEPENDENT TASKS


TASK A
TASK C
TASK B
C_1
C_2
C_{21}

(c) PARTITION EXPLOITING HIERARCHY

TASK OFF-LOADING

FIGURE 5


SCAN TRUNKS
SCAN NETWORK
SCAN LINES
SCANNING FUNCTIONS
PERFORM MAINTENANCE OPERATIONS
PROCESS SERVICE REQUESTS
SET UP CALL
TEAR DOWN CALL
TEST AVAILABLE PATH
SIGNAL NETWORK
IDLE PATH
UPDATE NETWORK MODEL

EXAMPLE: TELEPHONE SWITCHING PROCESSOR

FIGURE 6

SCHEDULING OF MULTICONFIGURABLE PIPELINES

A. THAMPY THOMAS
Intersil Incorporated
10900 N. Tantau Ave.
Cupertino, Ca 95014

EDWARD S. DAVIDSON
Coordinated Science Lab
University of Illinois
Urbana, Ill 61801

ABSTRACT

A pipeline is a collection of hardware segments which can operate simultaneously. The pipeline is reconfigured for each computation that flows through it. In pipelines with no internal buffering, two or more computations attempting to use a particular segment during the same time unit cause a collision. A simple shift register controller is developed to implement various collision-free sequencing strategies which compare favorably in performance with common practice. An optimum scheduling strategy is also developed for a stable task arrival pattern.

INTRODUCTION

A pipelined processor consists of several specialized subprocessors called segments. Each segment performs a specific part of a particular computation. Each single computational task must be known to require a particular sequence or use pattern of the segments in time. Any set of segment use patterns may be specified. Tasks are scheduled in an overlapped fashion so that many segments may be active at the same time.

For each computation, the segment use pattern must be fully known. The ease of obtaining such knowledge depends on the level at which pipelining is applied. For example, if the processor implements a floating point multiply task, the segment use pattern is easily obtained for a particular implementation. If, on the other hand, there are two segments a central processor and a peripheral processor, the segment use pattern for a task is generally difficult, if not impossible, to obtain.

A pipelined p-function generator can compute p distinct functions and hence it has p associated segment use patterns. Associated with each operand set to be initiated in the pipeline is a function-tag identifying which of the segment use patterns must be followed for its flow through the pipeline. An example of a multifunction unit is the arithmetic unit of Texas Instrument's Advanced Scientific Computer which has 7 hardware segments with about 20 possible configurations.

It is assumed that some pairs of computations use at least some of the same segments. In pipelines with no internal buffering, the condition of two or more computations attempting to use a particular segment during the same time unit is called a collision. Collisions may occur between two or more tasks with the same function tag. Two or more tasks with distinct function tags may also collide with each other. The control mechanism in pipelines with no internal buffers decides only when to initiate the next waiting task, so as to avoid collisions. Task flow cannot be delayed internally for collision avoidance.

A STATE MODEL FOR MULTIFUNCTION UNITS

A model for multifunction units can be developed which is independent of the specific calculations to be performed. The model indicates which segments of the multifunction unit are reserved in which time units after initiation, for each distinct function tag. Such a data flow model may be shown to furnish all the information necessary for collision-free sequencing of tasks in the pipeline.

The segment use pattern for a computation requiring the use of m segments in n units of time can be graphically represented by a mxn reservation table. The segments are listed with the same subscripts for each function tag, even if not all segments are used for each function tag. Row i column j of reservation table K has a K if and only if the computation for function K requires segment i during the j^{th} time unit after function K has been initiated. Several consecutive K's may be placed in the same row (a segment may take more than one time unit to complete its computational step) and several K's may be placed in the same column (more than one segment may be used in parallel by the computation). For pipelines with data flow that exhibits arbitrary feedback more than one K appears in some or all rows of the reservation table. Unused segments correspond to rows with no K's. The same time unit is used for all reservation tables so that the reservation tables may be overlaid.

Figure 1(a) shows the overlaid reservation tables for a 2-function unit with functions A and B. It can be seen that a previously initiated function A can collide with another function A issued 2 or 3 time periods later or with a function B issued 2 or 4 time units later. Similarly a previously initiated function B can collide with a function A initiated 1, 2 or 4 time units later or with another function B issued 2 or 3 time units later. It is assumed here that tasks are not initiated simultaneously.

The time between any two task initiations, not necessarily successive initiations, is the initiation interval. A forbidden initiation interval is then a collision-causing initiation interval. A task using the i^{th} reservation table will collide with a previously initiated task using the j^{th} reservation table if and only if the number of time units between initiations is a forbidden initiation interval for the ordered function pair <j,i>.

The cross collision vector V_{ij} is the set of forbidden initiation intervals for the ordered function pair <j,i>. The V_{ij} vector, $C_q \ldots C_2 C_1$, may be calculated by overlaying the i^{th} and j^{th} reservation tables. $C_r=1$, if any row of the overlaid table contains an i in column k and j in column k+r for some k. $C_r=0$, otherwise. Thus V_{AA}, in figure 1(b), has 1's in position 2 and 3 and V_{BA} in positions 2 and 4, similarly, V_{AB} has 1's in positions 1, 2 and 4 and V_{BB} in positions 2 and 3. The vector dimension, q, may be taken as the largest integer such that $C_q=1$, or such that $C_q=1$ for some ordered function pair <j,i>.

There are p^2 cross collision vectors associated with a p-function unit. One way to represent the set of forbidden initiation intervals of a p-function unit is to organize the p^2 cross collision vectors as p collision matrices, as shown in figure 1(c).

A simple shift register controller is used to implement any collision-free sequencing strategy. A p-function unit can be controlled by a bank of p shift registers, 1,2... p. Shift register i controls the initiations of function i. The bank of shift registers is initially cleared to all 0's. The control bits for function initiations are the rightmost bits of the shift register bank. Function i may be initiated at the beginning of the next time unit if the control bit of shift register i is 0. The shift registers shift one position to the right at the beginning of every time unit, with 0's introduced on the left. Immediately after the initiation of function j (and the accompanying shift), the matrix M_j is OR'ed into the shift register bank, the contents of shift register i being OR'ed with the collision vector V_{ij}, for $1 \leq i \leq p$.

OR'ing the cross collision vector V_{ij} with the contents of shift register i is necessary and sufficient to prohibit the collision of future initiations of function i with the function j, just initiated. The control bit of shift register i is thus 1 if function i, initiated at the next time unit, would collide with any one of the previously initiated tasks.

Latency is the time between two successive initiations. The shift register controller completely models the latency characteristics of the pipeline. The states of the shift register bank immediately after collision-free initiations are represented in the unified state diagram, figure 2. Each state is coded by the contents of the shift register bank. The interesting states are those immediately following a task initiation. All other states are simply right shifts of these states.

Each arc in the unified state diagram corresponds to the initiation of a task and is labeled ($f\ell$), where f stands for the function tag of the initiated task and ℓ, latency, is the number of elapsed time units since the previous initiation. Initial state i corresponds to the state of the shift register bank after the very first initiation of function i, since the shift register bank is initially cleared to all 0's. For the first initiation, latency is undefined, since no tasks were initiated before it. The null node, the all-zero state of the bank, does not appear in the state diagram.

Every node has an outbound arc for each 0 in its state coding. An arc corresponding to a 0 in position i,j is labeled ($f_i j$) signifying function f_i and latency j. In addition, there are p arcs leaving each node labeled (f_i, $(q+1)^+$), $1 \leq i \leq p$, where q is the maximum column dimension of any row in the state coding for the node. These p arcs are inbound to the p initial nodes, indicating that if more than q time units have elapsed before function i is initiated, the shift register bank returns to the state, represented by the collision matrix M_i. In complex state diagrams, these arcs are left out and shown, instead, by incoming arcs at the p initial nodes, labeled (f_i, $(q+1)^+$), $1 \leq i \leq p$. An arc labeled ($f\ell$) leaving a node with a state coding S leads to a node with the state coding S^1. S^1 is S shifted right ℓ positions and OR'ed with the collision matrix M_f.

The unified state diagram is a compact representation of all possible collision-free function-latency sequences of the multifunction unit. Cycles in the unified state diagram correspond to collision-free patterns of task initiations in the pipeline which can be maintained indefinitely. Cycles which have only functions of one type in their initiation sequences are called pure cycles. It can be shown that a cycle specified by its function-latency sequence, passes through only one set of nodes in the unified state diagram. Also, any cycle in the unified state diagram can be entered from at least one of the initial nodes along a path with the same function-latency sequence as that of the cycle itself. For example, the cycle (A3 B1) in figure 2 is reached by an arc labeled A3 from initial node B or arc B1 from initial node A. Such a path exists for each choice of initial arc from the cycle.

GREEDY STRATEGY

It is most straightforward to use the shift register controller to schedule functions, in the order in which they arrive at the input queue of the multifunction unit, at the earliest possible opportunity subject only to collision avoidance. Such a strategy is called greedy control.

Consider the 3-function unit and the partial unified state diagram shown in figure 3. Assume that functions arrive continuously at the input queue for the 3-function unit with an arrival pattern ACB. If greedy strategy is used, the function initiations follow a cyclic pattern A* (C4 B2 A5) with an average latency of 11/3=3.66. However, if functions may be taken out of order, a greedy cycle of A* (B1 C3 A5) is reached in which the functions are initiated in the order ABC. The average latency of this greedy cycle is 9/3=3.0, which is less than the average latency achieved by initiating the functions in the order ACB. So, occasionally higher throughput may be possible by initiating functions in something other than a first-in-first-out basis.

There are also instances in which higher performance is obtained by allowing a few initiation opportunities to pass by in a carefully controlled manner. As is clear from figure 3, there is a nongreedy cycle A* B1 C4 (A1 B1 C4) that has an average latency of 6/3=2.0. The nongreedy cycle (A1 B1 C4) is clearly better than the greedy cycles (C4 B2 A5) and (B1 C3 A5).

Since throughput is the reciprocal of the average latency, maximum throughput is achieved by an (optimum) scheduling strategy that achieves minimum average latency with no collisions. Although greedy control is not always optimal (there are pipelines for which no greedy cycle is optimal), it is the simplest strategy to implement with the shift register controller.

OPTIMUM STRATEGY

If the task arrival pattern is predictable and "sufficient" freedom in scheduling is allowed, a globally optimum sequencing strategy may be evolved which provides the maximum throughput. Given the task arrival pattern, the optimum control problem for a multifunction unit is to construct a collision-free pattern of function initiations which is compatible with the arrival pattern and achieves maximum throughput. The continuously arriving tasks are then executed by repetitive iterations of this pattern.

$<F_1, F_2 \ldots\ldots\ldots, F_p>$ is a function mix for a p-function unit if, for $1 \le i \le p$, $0 \le F_i \le 1$ and $\sum_{1 \le i \le p} F_i = 1$. Function mixes are used to indicate that in a particular sequence of initiations the fraction of initiations corresponding to function i is F_i.

The state diagram has one or more minimum average latency cycles. However, these limit the mix of function types which must be executed. For other mixes, it has been found that some linear combination of p or fewer "good" cycles achieve minimum average latency. Given the collision vectors, an efficient search procedure is developed to find these good cycles while typically constructing only a small portion of the state diagram. Given a desired function mix, the appropriate optimum cycle combination is found by solving a linear program.

Linear combination of simple cycles:

If in a single traverse of a cycle in the unified state diagram, each node is visited exactly once, the cycle is a simple cycle. Since the number of arcs in a unified state diagram is finite, there is only a finite number of simple cycles. If any node in the unified state diagram is visited more than once in a single iteration of the cycle, the cycle is a complex cycle. A complex cycle can follow any arbitrary closed path and hence there is an infinite number of them. Fortunately one does not have to deal with complex cycles explicitly since the function mix and average latency of any complex cycle is determined by its component simple cycles.

Assume that two simple cycles of function initiations, C_1= (A3 B1) and C_2= (B2 B3 B4) are available for sequencing a 2-function unit. By traversing (A3 B1) over and over, a function mix which has 50% A's and 50% B's is executed with an average latency of 2.0. Repetitive iterations of the cycle (B2 B3 B4) execute a function mix with 100% B's, with an average latency of 3.0. Linear combinations of these two cycles realize any function mix with 50% or more B's.

The span of a cycle set is the range of function mixes realized by linear combination of cycles from that cycle set. For example, the range of function mixes, achieved by the cycle set {(A3 B1), (B2 B3 B4)} is limited to 50-100% B's (or 50-0% A's).

A spanning cycle set can realize any desired function mix by an appropriate linear combination of its cycles. For a p-function unit p pure cycles are required for any spanning set, one for each function type. P pure cycles are necessary and sufficient to achieve any arbitrary mix of p functions. However, other cycles may be included in the spanning set.

Given the spanning set {(A1 A4), (B1 B4), (A3 B1)}, the function mix (F_A F_B) = (1/3 2/3) is realized by the cycle combinations (A1 A4) (B1 B4) (B1 B4) and (A3 B1) (A3 B1)·(B1 B4) with average latencies of 2.50 and 2.16 respectively. So, a mix may be achieved by more than one linear combination of cycles. The objective is then to choose the linear combination with the lowest average latency. Given a spanning cycle set and a function mix to be realized, there exists an efficient general linear programming formulation to determine the linear combination of cycles with the lowest average latency.

Good cycle set:

Given an arbitrary function mix, the cycle combination obtained by considering only cycles from an arbitrary spanning set may not realize the lowest possible average latency for that function mix. One way to ensure minimum average latency is to include all simple cycles of the unified state diagram in the spanning set. Since a complex cycle is a union of simple cycles, all possible complex cycles are implicitly included in such a spanning set. The linear programming formulation guarantees the best solution.

However, only a small percentage of the cycles are ever used in any optimum cycle combination, over all possible function mixes. So the spanning set for the linear program formulation need only include those cycles which will be used for some function mix. A good simple cycle is defined to be a simple cycle which achieves the minimum average latency for its own function mix. The good cycle set of a multifunction unit is then the set of all good simple cycles of that multifunction unit.

Given a simple cycle set S and an arbitrary path P in the unified state diagram, path P is redundant in S if and only if there exists a linear combination of cycles in S such that (i) the function mix of the cycle combination is the same as the function mix of the path P and (ii) the average latency of the cycle combination $\leq$ the average latency of the path P. Given a good cycle set S, a cycle C in S is a redundant good cycle if C is redundant in $S-\{C\}$. If redundant good simple cycles are eliminated from the good cycle set until no redundant cycles remain, the resulting set of cycles is known as a nonredundant good cycle set. There is a branch and bound algorithm to find a nonredundant good cycle set for a multifunction unit. The following theorems are necessary to justify the algorithm. Such an algorithm is possible because of the lead-in path structure for every cycle exhibited by multifunction units as stated at the end of the state model section.

Theorem - 1

A nonredundant good cycle set S for a p-function unit must contain exactly one optimum pure cycle for each of the p functions and possibly some nonpure cycles. Furthermore, S must be a spanning set.

Theorem - 2

Given an arbitrary function mix, there is a linear combination of p or fewer cycles from any nonredundant good cycle set S which achieves the function mix with the minimum average latency.

Theorem - 3

Given a set of cycles S and a cycle C not in S, if C is not redundant in S, then there exists an arc a_1 of C where $C=a_1 a_2 \ldots a_n$, such that, $a_1 a_2 \ldots a_i$, $1 \leq i \leq n$, is not redundant in S.

Algorithm (to find a nonredundant good cycle set S for a p-function unit):

Initialization: Set S to null. Function types are numbered 1,2...p. Initial node i, associated with function number i, is coded by the collision matrix M_i.

Step 1: Initial node 1 is selected as the current node. Set the path to the empty path.

Step 2: Among arcs outbound from the current node which have not yet been considered for this path, one with the minimum latency is selected.

If more than one arc has the same latency, one with the lowest function number is chosen. The selected arc is added to the path.

If none remain, go to step 8.

Step 3: Is the path nonredundant in S? If not, delete the last arc from the path and go to step 2.

Step 4: Set the current node equal to the end node of the path.

Step 5: Is the current node previously encountered in the path from the initial state? If not, go to step 2.

Step 6: Is the cycle formed nonredundant in S? If not, go to step 8.

Step 7: The cycle just found is a nonredundant simple cycle. Update the cycle set S to include the new cycle and delete redundant cycles from S, if any.

Step 8: If the path is empty with no initial states left, then terminate the algorithm.

If the path is empty, then set the current node to the next initial state and go to step 2.

Otherwise, reject the current node, delete the last arc from the path, set the new current node to the end node of the path and go to step 2.

The algorithm is easily modified in step 1 and given a head start by introducing any known good simple cycles into the initial cycle set. Minimum average latency cycles and optimum pure cycles are always good cycles.

Figure 4 shows the progress of the branch and bound algorithm to find a nonredundant good cycle set for the 2-function unit shown in figure 2.

Linear programming formulation to find the appropriate linear combination of cycles from a spanning cycle set to achieve the minimum average latency for any given function mix:

Minimize $\sum_{1 \leq j \leq n} x_j L_j$ subject to

$$\text{(i)} \quad \sum_{1\leq j\leq n} x_j F_{ij} = F_i, \qquad 1\leq i\leq p,$$

and

$$\text{(ii)} \quad x_j \geq 0, \qquad 1\leq j\leq n,$$

where:

p = the number of function types

n = the number of cycles in S

F_{ij} = fraction of the total number of initiations in which function i appears in cycle C_j

L_j = average latency of cycle C_j

F_i = function mix

($\sum_{1\leq i\leq p} F_i=1$, by definition).

x_j = fraction of the total number of arcs of the cycle combination which belong to cycle C_j

($\sum_{1\leq j\leq n} x_j = 1$, by definition).

To determine x_j^*, the relative number of traversals of cycle C_j used to complete a single traverse of the cycle combination, x_j is to be scaled by N_j, the number of arcs of cycle C_j.

$$x_j^* = x_j/N_j$$

The system of linear algebraic relations can be solved for the x_j by the well known techniques of linear programming.

Transient effects:

An underlying implicit assumption throughout the discussion on optimum cycles has been that the cycle combination obtained by linear combination of cycles from a nonredundant good cycle set forms a complex cycle. That is, the cycle combination can be traversed without the necessity of traversing any arcs in the unified state diagram which are not found in the cycle combination. However, the simple cycles that constitute a given cycle combination are not necessarily connected; i.e., they may not share a sufficient number of common nodes with one another.

It can be shown that connecting paths exist between unconnected simple cycles with the same function mixes as the cycles themselves. However, the average latency of a complex cycle formed of unconnected simple cycles may be greater than the average latency of some complex cycle that would have been possible if the simple cycles shared a sufficient number of common nodes with one another.

The degradation in performance because of the transient (connecting) behavior of a complex cycle set must be subject to a case by case analysis involving the arrival stability of the function mix with respect to the desired function initiation pattern and the length of the operand queues. The objective should be to minimize the number of transitions between the unconnected cycles.

CONCLUSION

Structured data machines, such as CDC STAR and TI ASC have machine language instructions to perform the same function mix on each component of a vector or matrix. The pipelines are configured so that tasks in execution at the same time must involve only a single function tag. If the operand queue for each vector operation is long, the sequencing strategy is to optimize the throughput for each function type. However, significant transients are then present when instruction types change. For short vectors the degradation in performance due to transient behavior may dominate any short term performance improvement by this strategy.

There are also situations when scalar and vector operations are intermixed. The arrival pattern for each vector operation is known to consist only of a single function. The incoming pattern of functions for scalar operations cannot be predicted in the short run. If a large number of operand sets are waiting to be initiated, the optimum control strategy based on the function mix present at the input queues gives near maximum throughput. The best general strategy is then to switch back and forth between greedy control for scalar operations and short vectors and minimum average latency control for one or more long vector operations or large queues of operand sets, waiting to be initiated.

In minimum average latency control, one must bear in mind that queue lengths may vary widely, since the initiation pattern realizes the given function mix only after each complete iteration of the complex cycle, formed from the optimum cycle combination for the function mix. Some function queues may even be empty when the complex cycle controller requests tasks from them. The initiation pattern required by the complex cycle may then not be realizable. Tasks are placed in function queues only after precedence ordering restrictions between the tasks have been resolved.

Minimum average latency control must then be looked upon as an upper bound to the performance achievable by a pipeline configuration. The transients and end effects can only be evaluated for a specific case and can be made arbitrarily small if sufficient freedom in scheduling is allowed. Simple greedy control, though not always optimal, generally provides higher throughput than the control procedures used in existing computer systems, employing multifunction units.

In contrast to traditional scheduling algorithms, the computational complexity of the algorithm presented here to generate the optimum scheduling strategy for multifunction units depends only on the number of distinct function tags in the set of tasks to be initiated, rather than on the number of tasks to be scheduled.

REFERENCES

1. E. S. Davidson, "The Design and Control of Pipelined Function Generators," Proc. 1971 Int. IEEE Conf. on Systems, Networks, and Computers, Oaxtepec, Mexico, Jan. 1971.

2. E. S. Davidson, "Scheduling for Pipelined Processors," Proc. Seventh Hawaii International Conference on System Sciences, Jan. 1974.

3. L. E. Shar, "Design and Scheduling of Statically Configured Pipelines," Tech. Report No. 42, DSL, Stanford University, Sept. 1972.

4. C. Stephenson, "Control of a Variable Configuration Pipelined Arithmetic Unit," Proc. Eleventh Annual Allerton Conference on Circuit and System Theory, Oct. 1973.

Reservation Tables:

	t_1	t_2	t_3	t_4	t_5
S_1	A	B		A	B
S_2		A		B	
S_3	B		AB		A

Figure 1a

Cross Collision Vectors:

$$\begin{array}{lcccc} & C_4 & C_3 & C_2 & C_1 \\ V_{AA} = & & 1 & 1 & 0 \\ V_{BA} = & 1 & 0 & 1 & 0 \end{array} \qquad \begin{array}{lcccc} & C_4 & C_3 & C_2 & C_1 \\ V_{AB} = & 1 & 0 & 1 & 1 \\ V_{BB} = & & 1 & 1 & 0 \end{array}$$

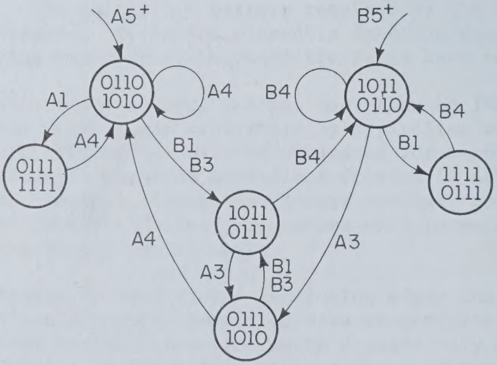
Figure 1b

Collision Matrices:

$$M_A = \begin{array}{ll} 0110 & (AA) \\ 1010 & (BA) \end{array} \qquad M_B = \begin{array}{ll} 1011 & (AB) \\ 0110 & (BB) \end{array}$$

FP-4229

Figure 1c



A5+
B5+
0110
1010
A4
A1
A4
0111
1111
B1
B3
B4
B4
1011
0110
B4
B1
1111
0111
1011
0111
A4
A3
A3
B1
B3
0111
1010

Unified State Diagram

Best Pure A Cycle (A1 A4) Av. Latency 2½
Best Pure B Cycle (B1 B4) Av. Latency 2½
Best Cycle (B1 A3) Av. Latency 2

FS-4062

Figure 2

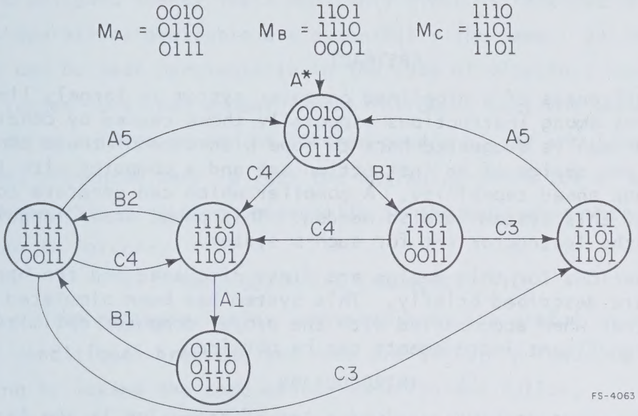

M_A = 0010 0110 0111
M_B = 1101 1110 0001
M_C = 1110 1101 1101
A*
0010 0110 0111
A5
A5
C4
B1
B2
1111 1111 0011
1110 1101 1101
1101 1111 0011
1111 1101 1101
C4
C3
C4
A1
B1
0111 0110 0111
C3
FS-4063

Figure 3. Nonoptimum Greedy Cycles

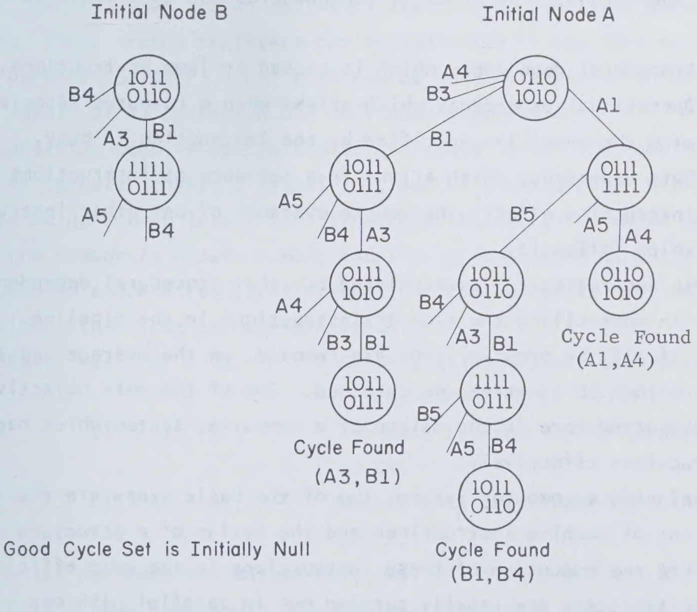

Initial Node B
Initial Node A
1011 0110
B4
A3
B1
1111 0111
A5
B4
A4
B3
0110 1010
A1
B1
1011 0111
0111 1111
A5
B4
A3
B5
A5
A4
0111 1010
1011 0110
0110 1010
A4
B3
B1
B4
A3
B1
Cycle Found (A1,A4)
1011 0111
1111 0111
B5
A5
B4
Cycle Found (A3,B1)
1011 0110
Good Cycle Set is Initially Null
Cycle Found (B1,B4)
FP-4230

Figure 4. Finding the Good Cycle Set with the Branch and Bound Algorithm

SUGGESTIONS FOR IMPROVEMENTS TO A PIPELINED COMPUTER SYSTEM

LAWRENCE CHEUNG AND FREDERIC J. MOWLE
Purdue University
West Lafayette, Indiana 47907

ABSTRACT

The effectiveness of a pipelined computer system is largely limited by the dependencies among instructions especially those caused by conditional jumps. An approach is suggested here to cope with some of these problems. This includes the design of an instruction set and a computer with jump instruction look ahead capability. A compiler which can generate code making full use of this system is also needed. This paper deals largely with the design of the Instructor set for such a system.

The motivations for this design are first discussed and the hardware requirements are described briefly. This system has been simulated and the results show that when accompanied with the proper computer optimization techniques, significant improvements can be obtained.

I. INTRODUCTION

Pipelined computers have received a lot of attention in the last decade and are regarded as one of the best ways to improve the computing power of a single processor. However, this concept suffers from the fact that instructions are not independent of each other, thereby limiting the number of concurrent operations. The types of dependencies can be classified as follows [1]:

(1) Procedural dependency which is caused by jump instructions.

(2) Operational dependency which arises when a resource associated with the operation specified by the instruction is busy.

(3) Data dependency which arises in a sequence of instructions when an instruction effects the source operands of any other instructions which follow it.

Riseman and Foster [2] have pointed out that procedural dependency is a key factor in controlling the flow of instructions in the pipeline. They claim that if all the program jumps are removed, on the average, an improvement of more than 50 times can be obtained. One of the main objectives of the work suggested here is the deisgn of a computing system which handles jump instructions effectively.

In designing a computer system, two of the basic steps are the selection of a set of machine instructions and the design of a structure which can carry out the execution of these instructions in the most efficient manner. These two steps are usually carried out in parallel with one modifying the other in the initial design phase. However, very little attention is given to show how these instructions will be actually used by computer programmers. Information about the kind of instructions needed to ease the burden of the programmer in his quest to produce concise and efficient code is seldom used during the design.

If one compares the instruction sets of present day machines with those which were designed twenty years ago, only minor differences can be observed. The basic operations available are essentially the same. Certainly some improvement can be made particularly in the case of pipelined computers.

The suggestions that appear in the following sections were arrived at after studying a large number of Fortran programs. It should be noted that the concept presented here is just the beginning and a lot of work remains to be done in this direction, especially in the case when the programs are not written in Fortran.

II. SOME DESIRABLE CHARACTERISTICS

Most of the characteristics suggested here are useful in reducing the effect of conditional branches and some are useful in reducing the burden of programming by making the code easier to write and follow.

(1) In order to minimize the effect of a conditional branch, the condition that determines the branch should be evaluated as early as possible and be placed in some readily accessable location.

This can be done by assigning a few special status registers for this purpose. Two types of status registers are needed, condition codes and switches. These status registers can be addressed in the same manner as a general purpose register, i.e., they can be used as operands in an operation and the result of an operation can be loaded into them.

A condition code, which is similar to those in IBM 360, can have three values, +1 (greater than), 0 (equal to), and -1 (less than). Whenever a number is being loaded into a condition code register, its value depends on whether the number is greater than, equal to or less than zero. Since it can be used to store a resultent operand, it can be set in many ways as shown below. The first two are equivalent.

COMPARE	C1	R1	R2,	Compare R1 and R2 and set condition code 1 (C1) accordingly.
SUB	C1	R1	R2,	Subtract R2 from R1 and send the result to C1.
MULT	C1	R1	R2,	Multiply R1 and R2 and send the result to C1.

A switch can have two values only, all 1's (TRUE, ON) and all 0's (FALSE, OFF). When the result of an operation is sent to a switch, it is turned OFF if the result is all 0; otherwise it is turned ON. A switch can also be set by the following type of instructions.

```
SGE    S1   R1   R2,            Set switch 1 if R1 is greater than or
                                equal to R2, otherwise reset switch 1.
SL     S1   R1   R2,            Set switch 1 if R1 is less than R2,
                                otherwise reset S1.
```

These properties are very useful in handling Boolean expressions and conditional jumps. This can be illustrated by the following example.

Consider these statements,

```
    LOGICAL   K
    K=A.GE.B.AND.B.EQ.C
```

Using the instructions shown above, the machine code generated would be

```
LOAD     R1   A,                Load variable A.
LOAD     R2   B,                Load variable B.
LOAD     R3   C,                Load variable C.
SGE      S1   R1   R2,          Set switch S1 if A is ≥ B.
SE       S2   R2   R3,          Set switch S2 if B = C.
AND      S1   S2   S1,          Set switch S3 if Both switch S1 and S2
                                are ON.
STORE    S1   K,                Set K equal to the value of switch S1.
```

Without those instructions, the code generated would probably have the following form.

```
     LOAD      R1   A,          Load A into R1.
     LOAD      R2   B,          Load B into R2.
     COMPARE   C1   R1   R2,    Compare A and B.
     ENTER     R4   0,          Enter 0 into R4.
     JL        C1   /L1,        Jump to /L1 if A < B.
     LOAD      R3   C,          Load C into R3.
     COMPARE   C1   R2   R3,    Compare B and C.
     JE        C1   /L2,        Jump to /L2 if B = C.
/L1  NEG       R4   R4,         Complement R4.
/L2  STORE     R4   K,          Store result in K.
```

Comparing these two approaches, the first one has the following advantages over the second one.

(i) It is simpler and more straight forward.

(ii) The code generated is shorter and more efficient.

(iii) No jump instruction is used in the first approach. This is especially important in the case of pipelined computers.

(2) To make full use of the status register suggested above, a powerful set of jump instructions is needed. Instructions similar to these below can be used.

JLE	C1	/L1,	Jump to /L1 if C1 has a value of 'less than' or 'equal to'.
JG	C1	/L1,	Jump to /L1 if C1 has a value of 'greater than'.
JST	(S1,S2)	/L1,	Jump to /L1 if both switch 1 (S1) and switch 2 (S2) are TRUE.
JSF	(S1,S2)	/L1,	Jump to /L1 if both S1 and S2 are FALSE.

The last two instructions are helpful whenever the decision to jump or not to jump depends on more than one variable. For example, consider the Fortran statement.

```
IF(A.EQ.B.AND.B.EQ.C)   GO TO /L1.
```

This can be converted into.

LOAD	R1	A,		Load variable A.
LOAD	R2	B,		Load variable B.
LOAD	R3	C,		Load variable C.
SE	S1	R1	R2,	Set switch S1 if A = B.
SE	S2	R2	R3,	Set switch S2 if B = C.
JST	(S1,S2)	/L1,		Jump to /L1 if Both S1 and S2 are TRUE.

(3) <u>Some frequently used functions should be incorporated into an instruction set.</u>

The following instructions are found to be useful in reducing the number of conditional jumps and/or in reducing programming efforts.

MVi	R1	R2	R3,	-	If switch i is ON, R2 goes to R1; otherwise, R3 goes to R1.
MAX	R1	R2	R3,	-	The maximum value of R2 and R3 goes to R1.
MIN	R1	R2	R3,	-	The minimum value of R2 and R3 goes to R1.
ABS	R1	R2,		-	Absolute value of R2 goes to R1.
FIX	R1	R2,		-	Convert R2 from floating point notation to fix point notation and put the result in R1.
FLT	R1	R2,		-	Convert R2 from fixed point notation to floating point notation and put the result in R1.
ETR	R1	R2	i j,	-	Extract i bits from R2 starting from bit j and load it into R1.
MCi	R1	R2	R3,	-	Mask both R2 and R3 by R1, treat them as unsigned integers and compare them, and set condition code C_i accordingly.

The first three instructions listed above are useful in eliminating jumps and also in simplifying expressions. The second three instructions are useful in numerical programming. The last two are powerful tools for data manipulations and table searching.

The following examples will illustrate how the first three instructions can be used. The concept used here is derived from dynamic algebra [3].

(i)

```
      IF(X.GT.Y)  GO TO 1
      T = X
      GO TO 2
    1 T = Y
    2 ...
```

This can be simplified into

```
LOAD    R1   X,            Load X into R1.
LOAD    R2   Y,            Load Y into R2.
MIN     R3   R1   R2,      Set R3 equal to the minimum of X and Y.
STORE   R3   T,            Store result in T.
```

(ii)

```
      X = A
      IF(W.GT.Z) X = B
```

This can be converted into

```
LOAD    R1   A,            Load variable A.
LOAD    R2   B,            Load variable B.
LOAD    R3   W,            Load variable W.
LOAD    R4   Z,            Load variable Z.
SG      S1   R3   R4,      Set S1 if W > Z.
MV1     R5   R2   R1,      Set R5 = R2 if S1 is ON; otherwise set
                           R5 = R1.
STORE   R5   X,            Store result in X.
```

(4) The system should be able to look ahead for jump instructions and locate the correct sequence of instructions as soon as possible in order to minimize the distrubance caused by the jump. A system capable of doing this is suggested in the next section.

(5) In a loop, the time spent waiting for data fetched from memory may be significant especially when the loop is short. This problem can be alleviated by using first in first out stacks to prefetch data.

The following properties are found to be desirable.

(i) The stacks should be addressed in the same way as general purpose registers.

(ii) When a stack is empty and is being referenced, its value is assigned to be the same as the last word loaded into the stack.

(iii) The first two elements in the stack can be referenced simultaneously. For example, the instruction, SUB R1 R21 R21, (R21 is assumed to be a first in first out stack) would cause the second element in the stack to be subtracted from the first element in the stack with the result being placed at R1. At the end of the operation, only the first element in the stack is lost and the second element becomes the first element.

The following example illustrates how a stack can be used.

```
      DO  1  I=1,K
1     N(I)=M(I)+M(I+1)
```

These state next can be converted into

```
    ENTER   R1   M,                    Load the based addresses of M and
    ENTER   R2   N,                    N into R1 and R2
    CLS     R21                        Clear stack R21
    ENTER   R3   1,                    Set I = 1
    INC     R4   R3   1,               Set R4 = R3 + 1
    LOAD    R5   K,                    Load K into R5
    LOAD    R21  R1   R3,              Load M(1) and M(2) into stack R21
    LOAD    R21  R1   R4,
/L1 SL      S1   R3   R5,              Set S1 if I < K
    INC     R4   R4   1,               Increment R4 by 1.
    LOAD    R21  R1   R4,              Prefetch M(I+2).
    ADD     R6   R21  R21,             M(I)+M(I+1)
    STORE   R6   R21  R21,             Store result in N(I)
    INC     R3   R3   1,               Increment I by 1.
    JST     S1   /L1,                  Go to /L1 if S1 is true.
```

Notice that in this example, S1 is set at the beginning of the loop and it is being tested only at the end of the loop, also M(I) and M(I+1) are fetched before they are actually used. All these maneuvers are to ensure a smooth flow of instructions and data to the execution unit of the processor.

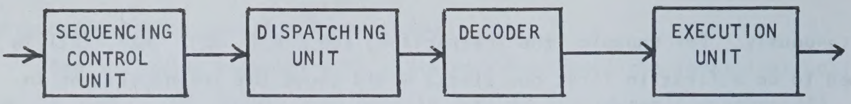
(6) Since several operations are being carried out simultaneously in a pipelined computer, the number of registers should be large enough to hold all the operands and temporary results.

It is found that a total of 32 registers are enough for most of the programs under consideration. These include 2 condition codes, 6 switches, 1 hard-wired zero, 2 first in first out stacks and 21 general purpose registers.

(7) In order to assure a smooth flow of instructions, extensive buffering schemes should be used at various stages of the computer structures.

III. A PROCESSOR WITH "JUMP" INSTRUCTION LOOK AHEAD CAPABILITY

The processor can be divided into four units, a sequencing control unit,



a dispatching unit, a decoder and an execution unit. The first three stages are responsible for fetching and decoding instructions and handling jump instructions. The execution unit is only responsible for carrying out the actual execution of instructions.

The sequencing control unit consists of an instruction buffer which acts as a small cache. Its main function is to look ahead for conditional and simple jumps and to execute them: i.e. to control the flow of instructions to the execution unit. It is also responsible for detecting loops, if there is a jump, and if the instruction is already in the buffer, no memory request will be issued.

The dispatching unit is responsible for extracting instructions from the buffer in the order indicated by the sequencing control unit and send them to the decoder.

The execution unit consists of different kinds of functional units and each is capable of performing a certain type of operations. This structure is similar to the one used by CDC 7600. However, in order to minimize operational dependency, the data handling algorithm [4] used in IBM 360/91 is incorporated into the design.

The decoder is responsible for the decoding of the instructions sent by the dispatching unit and for setting up appropriate flags and pointers so that the correct operands will be sent to the appropriate functional units and the result to wherever it is needed.

IV. CONCLUSION

A computer using the suggestions outlined in this paper has been simulated and several examples have been run. With proper compiler optimization techniques, on the average, more than two instructions can be executed in one machine cycle. For a 'conventional' computer, it takes several machine cycles to execute just one instruction. The number of jump instructions executed in these examples has been reduced by 50% when compared with the programs compiled using conventional methods.

The most significant result of this research is that it shows the need to improve the instruction set of a pipelined system and the importance of an efficient compiler for a pipelined computer. It should be pointed out here that the machine instructions suggested here are also very useful to a conventional computer.

REFERENCES

[1] Michael J. Flynn, "Some Computer Organization and Their Effectiveness", IEEE Trans. Computer, C-22, 8, pp. 752-761, Aug. 1973.

[2] E. M. Riseman and C. C. Foster, "The Inhibition of Potential Parallelism by Conditional Jumps", IEEE Trans. Computer, C-21, 12, pp. 1405-1411, Dec. 1972.

[3] R. Noonan, "Computer Programming With a Dynamic Algebra", Ph.D. Thesis, Department of Computer Science, Purdue University, 1971.

[4] R. M. Tomasuto, "An Efficient Algorithm for Exploiting Multiple Arithmetic Units", IBM Journal of Research and Development, Jan. 1967.

DYNAMIC MICROPROGRAMMING IN FORTRAN ENVIRONMENT

PHILIP S. LIU and FREDERIC J. MOWLE
School of Electrical Engineering
Purdue University
West Lafayette, Indiana 47907

ABSTRACT

Main memory requirement is greatly increased when high level language programs, such as Fortran programs, are compiled into microcode and buffered by a writable control memory through main memory. Bad locality of a microprogram may have a negative effect on the total execution time. Four alternate methods are proposed.

I. INTRODUCTION

Microprogrammed minicomputers are now available with a writable control memory in addition to the read only control memory which holds the target or end user machine's emulator. Several of these computers can support dynamic user microprogramming [1]. The writable control memory is usually left unused when high level language programs, such as Fortran programs, are being executed. This causes low utilization of a valuable resource. In this paper, four techniques are proposed which will allow certain parts of user Fortran programs to be compiled directly into microcode. All four of these techniques promise to speed up program execution.

II. PROGRAM EXECUTION WITH WRITABLE CONTROL MEMORY (WCM)

There are several different ways that a Fortran program can be executed with a writable control memory. The first method is to translate the entire Fortran program into microcode and to execute the resulting code directly from the writable control memory. This is of course the fastest method, but the extremely high cost of a large writable control memory makes this approach uneconomical and prohibitive. Flynn [2] has proposed a similar approach in which the writable control memory is divided into 2 parts. The first part is a 64-bit fast 4K writable control memory. The second part is a large 64-bit slow writable control memory (main memory) whose speed is ten times slower than the fast writable control memory (micromemory). However, the high cost of the main memory makes this method also unattractive. Flynn [2] also suggests that the most often used section of a Fortran program should be compiled into the fast storage with rest of the program being compiled into the slow memory. However this implies that statistics about the execution frequency of the program statements are known through previous runs. Ramamoothy [3] has proposed that such tactic should be adopted for production type programs. In this paper, no previous execution statistics about the user program will be assumed to be known. However, the user will be allowed to request specific portions of his program to be compiled into microcode if he so desires.

Recently Robbi [4] suggested that improved performance can be achieved if a program written in microcode is stored in main memory and buffered by a special purpose cache memory (or writable control memory). A disadvantage with this scheme is an increase in the amount of main memory required by a factor of three or four. This increase is due to the fact that it generally takes more than one microinstruction to carry out the steps dictated by one user main memory instruction . The number of bits in one microinstruction word are also usually greater than that in one main memory instruction word. Another problem with this scheme occurs when the user program displays bad locality or the buffer memory is small. In these cases the system spends most of its time swapping parts of the microprogram in and out of the buffer memory with little useful work being accomplished [5]. Wilkes [6] states that, in order for a gain in run time speed of 20% or 30% microprograms should only be moved into a fast memory (writable control memory) from a slow memory (main memory) when they will remain in the writable control memory for a relatively long time. In this paper, the inner loops of a Fortran program are proposed as the ideal program portions that should be microprogrammed and be overlayed in the writable control memory. Another motivation for this approach is due to Knuth [7] who has found that usually inner loops of a user Fortran program account for more than 60% of the execution time of the program, and that these inner loops constitude only 5% of the code.

The techniques proposed in this paper assume that there are two levels of writable memory in the system, the main memory and the writable control memory. Portions of a Fortran program, notably the inner loops, will be translated into microcode, and the rest of the program will be translated into the normal machine code. The object code in the main memory will be executed by the basic instruction set control sequence stored in a read-only control memory. The microcode generated for the Fortran program will be loaded into and be executed from the writable control memory. The program data is assumed to be stored in the main memory throughout the program execution. Using this approach, the main memory requirement is less than any of the other schemes suggested above, since the entire program is not translated into microcode. A gain in run time execution speed can also be expected, since now the inner loops of the Fortran program are to be executed using the fast writable control memory.

III. SELECTION AND TRANSLATION OF INNER LOOPS

In this section, several techniques for the selection and translation of inner loops in Fortran programs into microcode will be examined.

First Method: Static Loading of Inner Loops

In this method, a one pass compiler is to be assumed. During compilation, the compiler identifies inner loops in addition to generating main memory object code. As soon as an inner loop is identified, the compiler will go back and will convert the corresponding object code in the inner loop into microcode. Conversion and identification of inner loops stops when the WCM is full. From then on, the compiler will only generate main memory object code. During the conversion process, the constants used in the inner loop will be placed into the data field of the microinstructions where the constants will be used.

To convert the object code into microcode efficiently, the compiler will scan the object code sequentially and try to convert a number of object code lines into one microinstruction. The microinstruction can be assumed to have the following sample format.

ALU FIELD	TEST FIELD	TRUE NEXT ADDRESS	FALSE NEXT ADDRESS

In this format, the microinstruction is capable of performing in the same minor cycle one arithematic-logic operation on the registers specified by the ALU FIELD, test the result for some conditions specified by the TEST FIELD, and branch to one of two locations specified by the TRUE NEXT ADDRESS and the FALSE NEXT ADDRESS field depending on the outcome of the test. For example, the following three lines of object code can be converted into one microinstruction assuming the operands and the destination have been mapped into the host machine's registers:

LAC I   Load the accumulator with I
ADD J   Add J to the accumulator
STO K   Store result in K

I has been mapped into general purpose register 1 (GPR1)
J has been mapped into general purpose register 2 (GPR2)
K has been mapped into general purpose register 3 (GPR3)

Then the three object code lines can be converted into one microinstruction:

GPR1 ADD GPR2 → GPR3

The microcode produced will be reasonably efficient using the rule that as many lines of object code as possible be converted into one microinstruction.

During run time, before an inner loop is entered in the writable control memory, a microroutine is used to preload the necessary data into the host machine's registers. The main memory address of the necessary data can be passed to the microroutine using a standard software linkage method. Upon leaving the writable control memory, some data must be written back into main memory if their values have changed. This can be handled using the same linkage technique.

In the following, an analysis is carried out to estimate the run time overhead of preloading and restoring data. Let

N = Number of machine instructions in the loop.

F = Fraction of the N machine instructions that require a memory reference to fetch operands.

S = Fraction of the N machine instructions that require to store results back into the main memory.

M = Main memory cycle time.

P = Average execution time of the microinstructions in numbers of main memory cycle time, which preload an operand or restore a result. Main memory access time is not included.

T = Total overhead in preloading and restoring variables or operands.

Assuming all operands and results are single precision real numbers which occupy two main memory words each, the total overhad is:

$$T = 3MFN + 4MSN + (F+S)NPM$$

The first term is the time required to preload the operands. The second term is the time required to restore the results, and the third term is the total time spent in executing the restoring microroutine.

The total saving in execution time when the loop is implemented in microcode, comes from not having to fetch instructions, not having to store results back to main memory, not having to fetch operands, and not having to decode instructions. Letting

R_1 = The number of times the loop is executed.

G = Total saving in execution time.

D = Average decoding time for one machine instruction in numbers of main memory cycle time.

The total saving in execution time is given by the following expression:

$$G = MR_1N + 2MR_1FN + 2MR_1SN + MR_1DN$$

The first term is the total time saved by not having to fetch instructions, the second term is the time saved by not having to fetch operands or results already available in the host machine's registers, the third term is the total time saved by storing results into the host machine's registers and the fourth term is the total time saved in not having to decode additional instructions. For the total saving in execution time to exceed the overhead, it is required that

$$G > T$$

or $$R_1MN(1 + 2F + 2S + D) > (3F + 4S)NM + (F + S)NPM$$

$$R_1(1 + 2(F + S) + D) > (3F + 4S + P(F + S))$$

$$R_1(1 + 2(F + S) + D) \geq ((4 + P)\ (F + S))$$

Now $0 < F + S \leq 1$

In the worst case, when every instruction in the loop either loads or restores data, then $F + S = 1$ it is required that

$$R_1(1 + 2 + D) \geq (4 + P)$$

$$R_1(3 + D) \geq (4 + P)$$

or

$$R_{1min.} = \frac{4 + P}{3 + D}$$

The following table shows $R_{1min.}$ such that the total saving exceeds the overhead for different reasonable values of D and P:

P (Preloading and Restoring
Microroutine Execution Time)

D (Instruction Decoding Time)

$R_{1min.}$	0.5	1	2	3
0.2	1.40	1.56	1.88	2.20
0.5	1.28	1.43	1.70	2.00
1.0	1.12	1.25	1.50	1.75
2.0	0.90	1.00	1.20	1.40

Since the above analysis is a worst case analysis, it can be expected that in the narmal situation it takes a loop to execute approximately once or less to recover the preloading and restoring overhead.

Second Method: Static and Selective Loading of Inner Loops

In this technique, a two pass compiler is assumed. In the first pass, the compiler identifies inner loops in addition to generating main memory object code. While an inner loop is being compiled into object code, the starting location and the ending location of that section of code will be memorized by the compiler. It also assigns priorities to inner program loops according to the number of outer loops the inner loop has, the more outer loops an inner loop has the greater its priority. The size of each inner loop is also recorded.

The loops with the greatest gain in execution time are preferred to be translated into microcode if all the inner loops of a program cannot be fitted into the writable control memory. The following is a heuristic procedure which can be used to assign priorities to inner loops:

1. Detect the first backward jump statement or instruction in the main program or in the subroutine program.

2. Remember the entry label and the lables inside this inner loop if no other inner loops are embedded in it. Remember the labels encountered following this inner loop encountered later. Assign priority one to this inner loop and record its size.
3. Upon encountering another backward jump, check the label it jumps to, if it is one of those labels encountered following the last inner loop a new inner loop is found. Go to 2: otherwise go to 4.
4. If the jump to lable does not agree with any labels encountered after the last inner loop it is an outer loop, increase the priority of each loop within this outer loop by one and go to 3.

During the second pass, the inner loops with the greatest priority assigned to it will be converted into microcode . Within the same priority group, the inner loops with the largest size will be converted into microcode first until there is no more WCM space left.

Third Method: Overlay of Inner Loops.

In this method, inner loops are identified so that they can overlay each other in the writable control memory.

To overlay an inner loop in the writable control memory, the incoming inner loop microcode must be loaded into the writable control memory at run time in addition to preloading and restoring operands and results. This induces additional overhead. The loop is required to execute no more than C x L times in the WCM to recover this additional overhead, where

C = The average microinstructions required to implement one machine code assuming the required data is available in the host machine's registers.

L = Ratio of the length of one microinstruction word to one main memory word.

To use this technique, the following procedure will be carried out:

1). At compiler time, identify all inner loops and associate with each inner loop an identification number and convert each inner loop into microcode.

2). At run time when an inner loop is entered, its microcode is loaded into the WCM from main memory using a microloader. Its identification number is also saved. This number is checked each time an inner loop is to be loaded into WCM so that redundant loading of the same inner loop can be eliminated.

As a modification to the above procedure, all inner loops which have the same common outer loop may be assigned the same identification number and they will be loaded into WCM as one segment. In case an inner loop or segment is larger than the WCM space, part of the inner loop is assigned to execute from main memory.

Fourth Method: Execution of User Selected Code.

Based on intuition or detailed statistic gathering on his program, the user can determine which parts of his program execute more often than the other parts. To take advantage of this fact, one is allowed to specify which parts of his program should be translated into microcode, and the compiler will honor the user's request. Also the user can specify which portions of the chosen code should be translated into microcode first. For example:

```
.HELP 1
    .
STATEMENTS TO BE TRANSLATED INTO MICROCODE
    .
.END
```

And

```
.HELP 2
    .
STATEMENTS TO BE TRANSLATED INTO MICROCODE
    .
.END
```

The numbers 1 and 2 shown in .HELP specify which block of statements is to be translated into microcodes first. In this method, static loading of these codes into the control memory will be performed. The user may also request that the segments are to overlay each other.

IV. SUMMARY

The first two techniques in this paper do not require additional main memory, and the gain in execution speed is proportional to the size of the WCM. The third approach can give a considerable gain in execution speed with a small WCM no larger than 256 words. The increase in main memory requirement is modest when compared to the method when the whole program is to be translated into microcode. The last technique allows the user to have control over which parts of his program should be translated into microcode.

V. FURTHER WORK

A microprogrammed minicomputer has been designed to implement the above approaches efficiently. To observe the relative gain of the techniques, a simulator is being written to execute a set of Fortran programs. The WCM size and the main memory to the WCM speed ratio are two of the simulation parameters.

REFERENCES

(1) Agrawala A. K., and Rauscher T. G., "Microprogramming: Perspective and Status", IEEE Trans. on Computers, C-23 (August 1974), pp. 817-837.

(2) Tucker, A. B., and Flynn, M. J., "Dynamic Microprogramming: Processor Organization and Programming", Comm. of the ACM., 14 (April 1971) pp. 240-250.

(3) Ramamoorthy, C. V., and Tsuchiya, M., "A Study of User-Microprogrammable Computers", AFIPS Conference Proceedings, 36 (SJCC 1970), pp. 165-181.

(4) Robbi, A. D., "Microcache: A Buffer Memory Memory for Microprograms" Proceedings of the Sixth Annual IEEE Computer Society International Conference, San Francisco, September 1973.

(5) Denning, P. J., "The Working Set Model for Program Behavior", Comm. of the ACM, 11 (May 1968), pp. 323-333.

(6) Wilkes, M. W., "The Use of a Writable Control Store in a Multiprogramming Enviroment", Fifth Workshop on Microprogramming, Urbana, Illinois, September 1972 (Preprints).

(7) Knuth, D. E., "An Empirical Study of Fortran Programs," Software--Practice and Experience, 1 (1971), pp. 105-133.

NON SYMMETRIC PERMUTATIONS AND THE OPTIMAL BORDERED TRIANGULAR FORM OF A SPARSE MATRIX

A. SANGIOVANNI VINCENTELLI
Istituto di Elettrotecnica ed Elettronica del Politecnico
di Milano, Milano, Italy

ABSTRACT

In the analysis and design of large systems such as electronic circuit, a very sparse linear algebraic system of equation $Ax=b$ has to be solved repeatedly.
In order to efficiently solve the system mantaining the sparsity of the matrix, it is possible to use a particular partitioning strategy, i.e., a way to transform a non singular matrix with permutation into an optimal Bordered Triangular Form (B.T.F.). There was shown by Cheung and Kuh that the problem of transforming matrix A in the optimal B.T.F. with symmetric permutation is equivalent to the determination of a minimum essential set of a directed graph associated with A.
In this paper a graph theoretic interpretation is given to the problem of finding a non symmetric permutation on A to obtain an optimal B.T.F. This interpretation is obtained through decomposition theorems on nonsymmetric permutations and correspondence theorems between column (or row) permutations and topological operations on the directed graph representing A.

1. INTRODUCTION

It is well known that, at the moment, the solution of almost every large scale problems by computers (analysis, synthesis, optimization, etc.) involves the solution of a sparse system of linear algebraic equations /1/ . In the particular domain of large electrical network analysis and design, a large sparse system has to be solved many times /2/ .Thus it is very important to take full advantage of the sparse structure of the coefficient matrix of the system.

The most efficient way to solve such a system is Gaussian Elimination. This technique can be improved if a "reordering" of the coefficient matrix is carried out. A particular reordering strategy consists in reducing the matrix in the form shown in fig. 1, called Bordered Triangular Form (B.T.F.) /3,4/ .
If we take pivots in the natural order, the possibly generated fill-ins occur only in the shaded area and the storage of the matrix during the elimination process is simplified. Moreover this form is particularly well-suited for tearing /5/ and for inversion by bordering /6/ .

Given a matrix A in whatsoever form, it is possible to put it in different Bordered Triangular Forms in which A_{11} is of different size; obviously, among them, we are interested in the "optimal" one. It is assumed that the B.T.F. in which A_{11} is of largest size is optimum.

Then, the optimization problem can be stated as:

P1 - Given a matrix A, find two permutation matrices, P and Q, such that PAQ has the minumum number (k_{min}) of rows and columns which do not belong to the main block of the B.T.F. (A_{11})

$$k_{\min} = \min_{P,Q} k(P,Q) \qquad (1.1)$$

In /4/, it has been investigated the problem P1 with the restrictions:

A1) $Q=P^T$, i.e., symmetric permutations only are allowed
A2) the diagonal pivoting on A is stable in any order;

and a graph interpretation of this restricted problems has been given.

In this paper, a graph theoretic interpretation, fundamental for the development of efficient algorithms, to a more general case is given.

In fact, it is assumed only that :

B1 - The diagonal pivoting on PAQ is stable in any order.

2. GRAPHS AND MATRICES

A few elementary graph theoretical notions are recalled in this section. All undefined terms must be understood in the Harary's sense /7/. Let $G(X,U)$ $[G(X,E)]$ be a (di)graph with a set of vertices X and a set of undirected (directed) edges $U = \{\{x_i,x_j\} \,/\, x_i,x_j \in X\}$ $(E= \{(x_i,x_j)/x_i,x_j \in X\})$. A simple (directed) path $\mu(x_i,x_j)$ of lenghth ℓ is an ordered set of distinct vertices $\mu(x_i,x_j)=\{p_o,\ldots,p_k,\ldots,p_\ell\}$ such that $p_o = x_i$ and $p_\ell = x_j$, $\{p_k, p_{k+1}\} \in U$ $((p_k,p_{k+1}) \in E)$, for $k = 0,1,\ldots,\ell-1$. A simple cycle is a simple path in which $x_i = x_j$. The section (di)graph defined on a subset $Y \subset X$ is $G(Y) = G(Y, U(Y))$ $(=G(Y,E(Y)))$ where $U(Y)=\{\{x_i,x_j\} \in U \mid x_i,x_j \in Y\}$ $(E(Y) =\{(x_i,x_j) \in E \mid x_i,x_j \in Y\})$

Given a digraph G(X,E), the contraction of an arc (x_i,x_j) is accomplished by replacing nodes x_i and x_j by a single node x_k. An arc (x_k,x_h) is provided in the contraction graph for each arc (x_i,x_h) or (x_j, x_h) in the original digraph; the arc (x_i,x_j) is lost.

A contraction graph w.r.t. a subset $E^* \subset E$ is the graph obtained contracting all the edges in E^* .

A subset $S \subset X$ is an essential set of G(X,E) if the section digraph G(X-S) is acyclic, i.e., it has no cycles. An essential set of minimum cardinality is called minimum essential set and its cardinality is called the index of G. A bipartite (di)graph G(S,T,U) (G(S,T,E)) is a (di)graph G(X,U) (G(X,E)) such that $S \cup T = X$, $S \cap T = \emptyset$ and the section (di)graphs G(S) and G(T) are both vertex graphs . Given a bipartite (di)graph G(S,T,U) (G(S,T,E)) a matching I is a subset of U (E) such that no two edges in I are incident to the same node. A node $x \in X$ is said covered if there is an edge in I that incides in it. A complete matching is a matching such that all the nodes of the bipartite (di)graph are covered. Given a bipartite graph and a matching I, an alternating simple path $\lambda(x_i, x_j)$ is a simple path such that if the edges $\{p_k, p_{k+1}\}$ with k even are in I, the edges with k odd are not in I or viceversa. An alternating simple cycle is an alternating simple path $\lambda(x_i,x_j)$ in which $x_i = x_j$. It is well known that it is possible to describe the zero-non zero structure of a matrix via a graph.

In particular, we can associate to a matrix A (nxn) a bipartite graph as follows :

RA1 - The number of the nodes in S is equal to n.
They "represent" the rows of A.

RA2 - The number of the nodes in T is equal to n.
They "represent" the columns of A.

RA3 - In $G^A(S,T,U)$ there is an edge $\{s_i, t_j\}$ joining two vertices, s_i and t_j, if and only if there is a non zero element a_{ij} in A.

This representation can be used to determine for which variable each equation has to be solved during the elimination process. These variables are called output variables/5/.

If we want that the equations can be solved for their output variables at any time during the elimination process, we must require that the coefficients of the output variables in their equations are nonzeros.

It is immediate to see that the determination of a constrained output set is equivalent to the choice of a set of non zero coefficients in the coefficient matrix A, such that there is one and only one chosen coefficient in each row and in each column of A.

It has to be observed that a complete matching I_c in $G^A(S,T,U)$ is corresponding to a set of n coefficients with row and column indices different; then, the selection of a complete matching in $G^A(S,T,U)$ is equivalent to choose a costrained output set in A.

Usually /3,4,8,9/ the output variables are assumed as already chosen and they are reordered so that the number associated with their equation is coincident with their associated number, i.e., the coefficients of the output variables are set on the main diagonal of A by a column (or row) permutation.

Then, it is possible /4,7,8/ to associate a directed simple graph $G^A(X,E)$ to the structure of A (in which the main diagonal entries are the coefficients of the output set) as follows:

RB1 - The number of nodes in X is equal to n.
Each of them represents an equation and its output variable "together".

RB2 - In $G^A(X,E)$, there is an edge (x_i, x_j) joining two vertices, x_i and x_j, $(i \neq j)$ if and only if $a_{ij} \neq 0$.

If we consider a symmetric permutation on A, PAP^T, the graph G^{PAP^T} associated to PAP^T with the previous rules is isomorphic to G^A. In fact, it differs only for the number associated to the nodes.

The introduction of $G^A(X,E)$ allows us to give the following Proposition :

Proposition 2-1 /4/ The determination of an optimal Bordered Triangular Form of a given matrix A via symmetric permutations, is equivalent to the determination of a minimum essential set in its corresponding graph $G^A(X,E)$. □

Obviously, this formulation is possible because the symmetric permutations on A do not "change" the topology of the directed graph associated to A.

However, when we apply a nonsymmetric permutation on A, PAQ, the directed graph G^{PAQ}, corresponding to PAQ, has "different" topology.

Then, the determination of a minimum essential set of the graph associated to A looses its significance.

Therefore, a graph theoretic interpretation of the nonsymmetric "case" is not possible without a new insight into the problem.

3. GRAPH THEORETIC INTERPRETATION OF NON SYMMETRIC PERMUTATION

In this Section the goal of obtaining a graph interpretation to the more general case is achieved at first via the following theorem on nonsymmetric permutations :

Theorem 3.1 - Let A be a square non singular nxn matrix.
Let P and Q be two permutation matrices. It is always possible to decompose uniquely a non symmetric permutation of A, PAQ, into the application of two permutation : a column(+) per-

---

(+) The Theorems of this paper are concerned with column permutations. The same kind of Theorems and interpretation can be stated for row permutation as well (See /10/).

mutation AQ' and a symmetric permutation $P(AQ')P^T$(°).

□

In force of this Theorem, the optimization problem can be viewed as a two stages problem: in the first stage, it has to be found an optimal column permutation on A, AQ'; in the second one it has to be determined an optimal symmetric permutation on AQ', $P(AQ')P^T$. The assumption B1 implies that AQ' has non zeros on the main diagonal. A permutation Q' such that AQ' satisfies B1 is called a legal permutation. Q' can be considered as a graph operation on $G^A(X,E)$ to obtain $G^{AQ'}(X,E)$.

If we assume, without loosing generality, that A has non zero diagonal entries, A has a precise directed graph correspondent to it.

Then Q' can be viewed as a graph operation on $G^A(X,E)$ to obtain $G^{AQ'}(X,E)$. Thus, the first stage of the optimization problem is correspondent to the determination of a graph which has minimum index among all the possible graphs obtained from $G^A(X,E)$ via Q'

Now recall that a complete bipartite matching in $G^A(S,T,U)$ is correspondent to an output set in A. Then, the main diagonal entries of A are corresponding to a particular bipartite matching in $G^A(S,T,U)$ I_1. We can state a precise relation between complete bipartite matching, in $G^A(S,T,U)$ and legal column permutation AQ'.

Theorem 3.2. Let I_1 be the complete matching in $G^A(S,T,U)$ corresponding to the main diagonal entries of A. There is a one-to-one correspondence between the legal column permutation on A, AQ' and the change of I_1 with another complete bipartite matching I_2 in $G^A(S,T,U)$.

□

In order to determine all legal Q's, we have to individuate, by Theorem 3.2, all the possible different complete matchings in $G^A(S,T,U)$.

In the following Theorem we shall prove that all the possible bipartite complete matchings can be generated with simple operations from one of them.

Theorem 3.3. Given a bipartite graph $G(S,T,U)$ and a complete bipartite matching I_1, any other possible bipartite complete matching in $G(S,T,U)$, I_h, can be obtained from I_1 as follows:

R1) individuate one or more disjoint simple alternating cycles (e.g.m) C^js w.r.t. I_1. Let I'_1 be the set of edges in I_1 but not incident in the nodes of the alternating cycles, and $\bar{I}$ the set of edges incident in the vertices of the alternating cycles but not in I_1. Then

$$I_h = I'_1 \cup \bar{I}. \tag{3.1}$$

□

Now, we have found a precise relation between $G^A(S,T,U)$ and the column permutations on A. Let us observe that, if bipartite graphs are a more complete representation of A than directed graphs, as we have seen, $G^A(S,T,U)$ is of higher dimension and some simple interpretation of elimination procedures, partition techniques, etc., are more complex in these graphs. Then, our aim is to use as long as possible $G^A(X,E)$ or any way, graphs strictly related to it. We shall show in the sequel that this goal is achievable. The first step in this process is to find a relation between $G^A(S,T,U)$ and $G^A(X,E)$. This relation will be found as a particular case of a more general Theorem.

(°) The proof of this Theorem as well as the proofs of Theorems 3.2, 3.3, 3.4, and Lemma 3.1 are omitted because of limitation of space. They can be found in /10/.

Theorem 3.4. Let $G^A(S,T,U)$ be as in Theorem 3.3. Let $G^A(S,T,\overline{E})$ be the bipartite directed graph obtained from $G^A(S,T,U)$ orienting all the edges U from S to T. Let Q' be a legal column permutation matrix and I the complete matching in $G^A(S,T,\overline{E})$ correspondent to the main diagonal of AQ'. The graph $G^{AQ'}(X,E)$ obtained from AQ' with RB1, RB2 is the contraction graph of $G^A(S,T,\overline{E})$ w.r.t. I, $G_c(X,E)$.

□

It is also possible to find a precise correspondence between alternating cycles in a bipartite graph w.r.t. a complete matching I and directed cycles in its contraction graph w.r.t. I, as shown in the following Lemma.

Lemma 3.1 - Let G(S,T,U) be a bipartite graph and I a complete matching. Let $G(S,T,\overline{E})$ as in Theorem 3.4. Let $G_c(X,E)$ be the contraction graph of G(S,T,U) w.r.t. I. There is a one-to-one correspondence between alternating cycles w.r.t. I in G(S,T,U) and directed cycles in $G_c(X,E)$.

□

We are now able to prove the fundamental Theorem upon which the graph theoretic interpretation of the problem is based.

Theorem 3.5. Let $G^A(X,E)$ be the directed graph corresponding to a matrix A, according to R1 and R2. There is a one-to-one correspondence between the legal column permutations on A, AQ's, and the directed graphs $G^*(X,E)$ obtained from $G^A(X,E)$ with the following operations :

O1) chosen one or more (e.g., m) simple directed cycles in $G^A(X,E)$:

(3.2) $$\overline{C}^j = (p_1^j, \ldots, p_k^j, \ldots, p_{\ell_j}^j) \qquad j = \{1, \ldots, m\}$$

reverse their orientations in $G^*(X,E)$, from clockwise into counterclockwise (or viceversa), i.e., for each edge (p^j_{k-1}, p^j_k) in $\overline{C}^j$, substitute an edge (p^j_k, p^j_{k-1}) so that in $G^*(X,E)$ there is a cycle :

(3.3) $$\overline{C}^{j*} = (p^j_{\ell_j}, \ldots, p^j_k, \ldots, p^j_1)$$

O2) for each edge (x_h, p^j_k) in $G^A(X,E)$ s.t. $x_h \neq p^j_{k-1}$, i.e., for each edge incident in a node of $\overline{C}^j$, substitute an edge (x_h, p^j_{k-1}) in $G^*(X,E)$.

Proof.

I) Let Q' be a legal permutation matrix. By Theorem 3.2, there is a correspondence between changes of matchings in $G^A(S,T,U)$ obtained from A with RA1, RA2, RA3, and Q'. Let I_1 be the matching correspondent to the main diagonal of A and I_2 as in Theorem 3.3.

By Theorem 3.3, the change of matchings happens "along" alternating cycles

(3.4) $$C^j = (c_0^j, c_1^j, \ldots, c^j_{\ell_j}) \qquad j = \{1, \ldots, m\}$$

in $G^A(S,T,U)$. Let $G^A(S,T,\overline{E})$ be as in Theorem 3.4 By this Theorem, the directed graph $G^A(X,E)$ corresponding to A is the contraction graph of $G^A(S,T,\overline{E})$ w.r.t. I_1 and $G^{AQ'}(X,E)$ corresponding to AQ', is the contraction graph of $G^A(S,T,\overline{E})$ w.r.t. I_2. By rules RA and RB, the nodes $x_i \in X$ in the contraction graph are numbered as the nodes s_i in the edges (s_i, t_k) belonging to the complete matchings I_1 and I_2.

By Theorem 3.3, I_2 differs from I_1 only for the edges in the cycles C^j_s and then

(3.5) $$G^A(X') = G^{AQ'}(X')$$

where

(3.6) $$X' = \left\{ x_i \mid x_i \text{ "generated" by } (s_i,t_k) \in I_1 \wedge s_i,t_k \notin C^j{}_s \right\}$$

i.e., the section graphs of $G^{AQ'}(X,E)$ and of $G^A(X,E)$ w.r.t. the vertices not "corresponding" to the cycles C^j s are equal.
Let

(3.7) $$\bar{C}^j = (p_1{}^j, \ldots, p^j{}_k, \ldots\, p^j_{\ell_{j/2}}) \qquad j = \left\{1,\ldots m\right\}$$

be the directed simple cycles in $G^A(X,E)$ corresponding to the C^j s by Lemma 3.1, with

(3.8) $$p^j{}_{k+1} = \bar{c}^j{}_{2k,2k+1} \qquad k= \left\{0,\ldots, \frac{\ell_j -2}{2}\right\} \quad j= \left\{1,\ldots, m\right\}$$

and

(3.9) $$p^j_{\ell_{j/2}} = p^j{}_1 \qquad j= \left\{1,\ldots, m\right\}$$

In $G^{AQ'}(X,E)$ we consider the nodes corresponding to the cycles C^js. Because, according to Theorem 3.3,

(3.10) $$\bar{I} = \left\{c^j{}_{2k+1}\, ,\, c^j{}_{2k+2}\right\} \qquad k = \left\{0,\ldots, \frac{\ell_j -2}{2}\right\} \quad j = \left\{1,\ldots, m\right\}$$

and, by Theorem 3.4, we have

(3.11) $$c^j{}_{2k+1} \in S,\ c^j{}_{2k+2} \in T \longrightarrow \bar{c}^{*\,j}_{2k+1,\ 2k+2}$$

If we consider as correspondent in $G^A(x,E)$ and $G^{AQ'}(X,E)$ the

(3.12) $$\bar{c}^{*\,j}_{2k+1,2k+2} \longleftrightarrow \bar{c}^j_{2k,2k+1} \longleftrightarrow p^k_{k+1} \qquad k = \left\{0,\ldots, \frac{\ell_j -2}{2}\right\}$$

the set of nodes

(3.13) $$\bar{C}^{*\,j} = (p^j_{\ell_j/2}, \ldots, p^j_k, \ldots, p^j_1) \qquad j = \left\{1,\ldots m\right\}$$

are cycles; in fact :

(3.14) $$(p^j_k,\ p^j_{k-1}) \in E \qquad k = \left\{1,\ldots, \ell_j/2\right\} \quad j = \left\{1,\ldots, m\right\}$$

Moreover, we can observe that by definition of $G^A(S,T,E)$ and by (3.12) there is an edge $(x_h,\ p^j{}_k)$ in $G^{AQ'}(X,E)$ for each edge $(x_h,\ p^j{}_{k+1})$ (with $x_h \neq p^j{}_k$) "derived" from the edge $(s_h,\ \bar{c}^j{}_{2k})$. Thus the first part of the theorem is proved.

II) Let us consider a graph $G^A(X,E)$ representing a matrix A and a graph $G^*(X,E)$ obtained from $G^A(X,E)$ with O1 and O2. Let us consider the bipartite graph $G^A(S,T,\bar{E})$ as obtained from $G^A(X,E)$ in the following way :

(1) for each node $x_i \in X$, substitute a node $s_i \in S$ and a node $t_i \in T$, $i = \left\{1,\ \ldots,\ n\right\}$

(2) a) join each s_i and t_i with an edge $(s_i,\ t_i)$

b) for each edge $(x_h, x_k) \in E$ substitute an edge (s_h, t_k)

Now, let us consider the bipartite graph G^* $(S^*, T^*, \bar{E}^*)$ obtained from G^* (X,E) with the same rules previously introduced. If the $\bar{C}^j$s are the cycles along which the transformations O1 and O2 has been done, it is obvious that :

$$G^A(X') = G^* (X') \tag{3.15}$$

where

$$X' = \left\{ x \mid x \in X \wedge \notin \bar{C}^j, \forall j = \left\{1, \ldots, m\right\}\right\} \tag{3.16}$$

Then, by rules (1) and (2), the section graphs of $G^A(S,T,\bar{E})$ and of G^* $(S^*; T^*; \bar{E}^*)$ w.r.t. the nodes derived from X', are equal.
Let $S^j \subset S$ and $T^j \subset T$ be the sets of nodes in $G^A(S,T,\bar{E})$ "generated" by the nodes in the cycles $\bar{C}^j$ s and $S^{*j} \subset S$ and $T^{*j} \subset T^*$ the sets of nodes in G^* $(S^*, T^*, \bar{E}^*)$ generated by the corresponding nodes in the cycles $\bar{C}^{*j}$ s.
By O2, there is an edge (x_h, p^j_{k-1}) in G^* (X,E) for each (x_h, p^j_k) in $G^A(X,E)$ and, then, by (2) there is an edge $(s^*_h, t^{*j}_{p_{k-1}})$ in $G^*(S^*, T^*, \bar{E}^*)$ for each $(s_h, t^j_{p_k})$ in $G^A(S,T,\bar{E})$.

Moreover, by rules (1) and (2), and by O1,

(3;17) $$(s^j_{p_k}, t^j_{p_k}) \in \bar{E}$$

$$(s^j_{p_k}, t^j_{p_{k+1}}) \in \bar{E} \tag{3.18}$$

and

$$(s^{*j}_{p_k}, t^{*j}_{p_{k-1}}) \in \bar{E}^* \tag{3.19}$$

$$(s^{*j}_{p_k}, t^{*j}_{p_k}) \in \bar{E}^* \tag{3.20}$$

The one-to-one correspondence between S and S^*, T and T^* :

(3.21) ca) $s_i \longleftrightarrow s^*_i$ $i = \left\{1, \ldots, n\right\}$

(3.22) cb) $t_i \longleftrightarrow t^*_i$ if t_i and t^*_i are derived from X'

(3.23) cc) $t^j_{p_k} \longleftrightarrow t^{*j}_{p_{k-1}}$ otherwise

is an isomorphism, because for each edge joining two vertices in $G^A(S,T,\bar{E})$, there is an edge joining the corresponding vertices in $G^*(S^*, T^*, \bar{E}^*)$ and viceversa.

The isomorphism between $G^*(S^*, T^*, \bar{E}^*)$ and $G^A(S,T, \bar{E})$ implies that the matrix whose structure is represented by $G^*(S^*, T^*, \bar{E}^*)$ is a permutation of A. Because of ca), cb) and cc), this permutation is a column permutation AQ'. Q' can be obtained by the renumbering of nodes in T according to cb) and cc). The proof is completed.

$\Box$

The Theorem previously proved gives a precise description of the operations needed to generate all the graphs of the class defined by the legal column permutations.

In force of this result the optimization problem P1 with restriction B1 has the following complete graph interpretation :

PGR - 1 : Find a graph with minimum index among all the graphs generated from a graph $G^A(X,E)$, associated to a given matrix A, with O1 and O2 of Theorem 2.

2 : Determine a minimum essential set of the graph individuated in 1.

4.CONCLUSIONS

In this paper, the problem of determining an optimal Bordered Triangular Form (B.T.F.) of a given matrix has been investigated. This problem arises from the applications of sparse matrices techniques to large systems (in particular electrical networks) analysis and design.

It has been shown in /4/ / that Graph Theory can be profitably used in the determination of an optimal B.T.F. when only a certain class of transformations are allowed : symmetric permutations.

A generalization of this approach to the case in which also non symmetric permutations are allowed has been proposed as an open question in /4/ .

In this paper, the question has been partially answered: a graph theoretic interpretation has been given to the problem, but the Algorithms to solve this problem have not been discussed for limitation of space. However it is possible to note that an efficient Algorithm has not yet been found. An Algorithm based on bipartite graph is until now the best result obtained: it will be described in a forthcoming paper.

The results obtained in this paper allow us to propose new research direction as :

(i) Determination of an efficient algorithm based on Branch and Bound techniques, which is able to find a graph with minimum index in a class individuated by O1 and O2 of Theorem 3.5.

(ii) Individuation of heuristic rules for the selection of a graph with "good" index in the class previously defined, without excessive computation time and storage requirement in order to solve real problems in an effective way.

(iii) study of the properties of the class of graphs previously defined.

(iv) Theoretical consequences of the graph theoretic interpretation of non symmetric permutations in pivot strategies used within Gaussian Elimination.

BIBLIOGRAPHY

/1/ Himmelblau D.M. (Ed) "Decomposition of Large-Scale Problems", North-Holland Publ. Comp., 1973.

/2/ Hachtel G.D., Brayton R.K., Gustavson G.F. "The Sparse Tableay Approach to Network Analysis and Design", IEEE Trans. Circuit Theory, 1971, Vol. CT-18, n. 1.

/3/ Dickinson, J.C. "Finding Permutation operations to produce a large triangular submatrix", National Meeting, ORSA, Houston, Texas, 1965.

/4/ Cheung L.K., Kuh E.S. "The Bordered Triangular Matrix and Minimum Essential Sets of a Digraph", MEMO ERL-M 375, University of California, Feb. 1973. To appear IEEE Trans. Circuits and Systems, 1974

/5/ Steward D.V. "Partitioning and tearing system of equations", J. SIAM Numer. Anal., Sec. B;, Vol. 2, n. 2, 1965.

/6/ Faddeeva V.N. "Computational Methods of Linear Algebra" Dover Publications, Inc. N.Y., 1963.

/7/ Harary F. "Graph Teory", Addison-Wesley, Reading, Mass. 1969.

/8/ Rose D.J. "A Graph-Theoretic study of the Numerical Solution of Sparse Positive Definite System of Linear Equations", in Graph Theory and Computing, R.C. Read (ed.) Academic Press, New York 1971.

/9/ Ogbuobiri E.D., Tinney W.F., Walker J.W. "Sparsity - Directed Decomposition for Gaussian Elimination on Matrices, IEEE Trans. Power Apparatus and Systems, 1970, Vol. PAS-89, n. 1

/10/ Sangiovanni-Vincentelli A. "An Optimal Bordered Triangular Form of a Sparse Matrix" Memo LC-74/4 - Politecnico di Milano, 1974.

/11/ Guardabassi G. "A note on minimal essential set", IEEE trans. Circuit Theory, vol. CT-18, Sept. 1971.

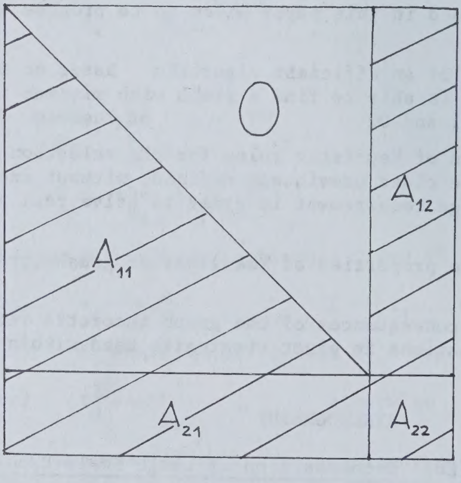


Fig. 1

The Bordered Triangular Form

NONLINEAR TRANSIENT ANALYSIS PROGRAMS: A COMPARISON

GEORGE W. ZOBRIST
University of South Florida
Tampa, Florida

ABSTRACT

This paper compares four nonlinear transient analysis programs. The programs are: IMAG2[1] (France), ITRAC[2] (U.K. and U.S.A.), NAP-2[3] (Denmark) and SCEPTRE-II[4] (U.S.A.). The networks used for comparison are: (1) RTL Inverter (with and without large time constant spread), and (2) five cascaded RTL Inverters (with and without large time constant spread).

A brief synopsis of the major features and analytical techniques involved in each program are given in Table I.* The output (see Table II)*, data presented is: (1) total CPU time, (2) number of integration steps.

GENERAL ANALYTIC TECHNIQUES DESCRIPTION

The basic analytic method is to formulate the nonlinear state equations using sparse matrix techniques, then use a version of the Newton-Raphson iteration to find the initial conditions, and then use an implicit integration routine to find the time domain solution.

The various programs use different approaches to the basic technique outlined above.

The programs included have the capability for using a continuous model for the nonlinear devices, either user supplied, or a fixed model.

INITIAL CONDITIONS

The Newton-Raphson[5] technique is the most widely used iterative technique used to find the initial condition solution of a set of nonlinear differential equations.

The procedure is to form equations describing the network as

$$\underset{\underline{\text{nonlinear}}}{g(i,v)} \qquad = \qquad \underset{\underline{\text{linear}}}{[H] \begin{bmatrix} i \\ v \end{bmatrix}} \, .$$

Then form an error function

$$f(i,v) = [H] \begin{bmatrix} i \\ v \end{bmatrix} - g(i,v)$$

---

*In some cases the information is incomplete.

and perform a linearization

$$f(i,v) \simeq f_i(i,v)_i + J_i\,[f(i,v)]\,[{}^{\Delta i}_{\Delta v}]_i\ .$$

This results in the following equation to solve for $[{}^{\Delta i}_{\Delta v}]_i$,

$$J[f(i,v)]_i\ \ [{}^{\Delta i}_{\Delta v}]_i = -f(i,v)_i\ .$$

The solution is then updated in the following manner

$$[{}^{i}_{v}]_{i+1} = [{}^{i}_{v}]_i + \alpha[{}^{\Delta i}_{\Delta v}]_i$$

and continued until

$$[\Delta i \Delta v]_i\ [{}^{\Delta i}_{\Delta v}]_i \leq E^2,$$

where $\alpha = 1$.

In general, convergency is not assured, divergency may occur and/or wrong solution may result. The advantages are that it is simple to implement and possesses quadratic convergency.

Divergence can be circumvented by choosing α so as to minimize $f(i,v)$, or at least to guarantee that $f(i,v)_{i+1} < f(i,v)_i$; i.e., by not taking a full step ($\alpha < 1$).

When nonlinearities of type e^x occur, there is a possibility of machine overflow; if $x > 0$. This can be circumvented by utilizing the specific nonlinearity in iteration process.

The convergency and wrong solution problem is circumvented by a "good guess" for a starting point, or the use of more sophisticated methods.

The various programs use one or more of the above modifications for initial condition determination.

TRANSIENT ANALYSIS

Implicit integration[5,6] is the standard technique used in the second generation transient analysis programs.

The general form for multistep implicit integration is:

$$X_{i+1} = -h\dot{X}_{i+1} - \sum_{j=1}^{k} \alpha_j X_{i-j+1}$$

The choice of h, k, and α's result in various regions of stability and solution times. It is in this regard that the various programs differ. In general, the use of an implicit integration scheme requires the solution of an auxiliary set of equations for $\dot{X}_{i+1}$. The introduction of sparse[5] matrix techniques has made implicit techniques tractable.

Implicit integration techniques are useful in solutions of systems of equations with large time constant spread.

NETWORK DESCRIPTION

The RTL inverter was used as the test network (see figure 1(a)). In figure 1(b), a comparatively small time constant was obtained by inserting the noted capacitor. The dependency of solution time upon the number of state variables involved was determined by cascading 5-RTL inverters (see figure 1(c)). In figure 1(d), a small capacitor was added to give a time constant spread.

The results were obtained for real circuit times from 100 to 4100 nsec (see figure 2). In figure 2, the input and output waveforms are shown.

REFERENCES

1. "IMAG2 - Un Programme de Simulation de Circuits Electriques", Jacolin, LeFaou, Pimort, and Veran, University of Grenoble, France (1970).
2. "Private Correspondence on ITRAC", R. W. Brown, Time Sharing Ltd., London, England.
3. "NAP-2: A Nonlinear Analysis Program for Electronic Circuits, Users Manual", T. Riibner-Peterson, Technical University of Denmark, Lyngby, Denmark (1973).
4. "Automated Digital Computer Program for Determining Responses of Electronic Circuits to Transient Nuclear Radiation, Volume II", H. M. Mathers, S. R. Sedore, and J. R. Sents, AFWL, Kirtland AFB, New Mexico (1967).
5. Computer Aided Network Design , D. Calahan, McGraw-Hill, Revised Ed (1972).
6. "Numerical Integration of Stiff Ordinary Differential Equations", C. W. Gear, University of Illinois, Urbana, Illinois (1967).

ACKNOWLEDGEMENTS

Partial support for this research was granted by the Science Research Council of Great Britain while the author was on leave of absence to the Computer Aided Design Project, University of Edinburgh, Edinburgh, Scotland. Acknowledgement is also made to Mr. C. LeFaou, University of Grenable, France, Mr. R. W.Brown, Time Sharing Ltd., London, England and Dr. E. Lindberg, Technical University of Denmark, Denmark, for running the example problems using IMAG2, ITRAC and NAP-2, respectively. ITRAC is a proprietary program of Berne Electronics, Inc., White Plains, New York.

PROGRAM NAME	IMAG2	ITRAC	NAP-2	SCEPTRE-II
Computer	IBM 360/67	PDP 10	IBM 370/165	IBM 360/65
Programming Language	Fortran	-	Fortran	Fortran
Memory Capacity	150K-250K	-	104K	182K
Interaction	Yes	Yes	No	No
Batch-Time Share	Yes	Time Share	Batch	Batch
Network Formulation	State Variable	State Variable	State Variable	State Variable
Model Capability	User	Fixed	User and Fixed	User
Sparse Matrix	-	Yes	Yes	Yes
Newton-Raphson	α for minimum f(i,v)	-	User Supplied α	Standard
Integration Routine	Implicit	Implicit	Implicit	Implicit
Step Size (h) and Order (k) Control	Step Size Variable and k = 2	-	Variable	Variable
Functional Routines	-	-	Yes	Yes
Sensitivities	DC	No	DC-Adjoint Tran.-Pertubation	No
Optimization	No	No	Yes	No
Origin	France	U.K. & U.S.A.	Denmark	U.S.A.

TABLE I - PROGRAM SYNOPSIS

T_F FINAL TIME SEC	CPU TIME** SEC				NO. OF INTEGRATION STEPS			
EXAMPLE	1	2	3	4	1	2	3	4
IMAG2								
100E-9	5.43	6.08	17.17	18.9	110	116	122	115
2100E-9	7.06	7.5	38.74	43.17	201	234	361	364
4100E-9	7.26	8.44	50.18	49.95	289	309	519	429
ITRAC								
100E-9	5.12				19			
2100E-9	3.80				80			
4100E-9	18.96	5.14	54.44	83.	60	60	224	214
NAP-2*								
100E-9	1.26	1.37	7.07	7.35	105	105	187	187
2100E-9	2.74	2.96	18.89	20.06	316	319	579	585
4100E-9	3.78	3.78	25.55	26.09	386	396	783	746
SCEPTRE-II*								
100E-9					131	403	290	554
2100E-9					1290	2421	1780	2593
4100E-9	35.40	103.70	261.49	280.75	2373	4096	3044	4106

*TRUNCATION ERROR
NAP-2 - REL - 1.E-3
SCEPTRE-II- ABS -.2E-3/7.5E-3
REL 0/0

*TIME STEP (MAX/MIN)
900E-9 / 100E-9
E-2 x T_F / E-5 x T_F

** CPU TIME
SCEPTRE-II - CPU TIME FOR ALL THREE T_F.

TABLE II - PROGRAM DATA

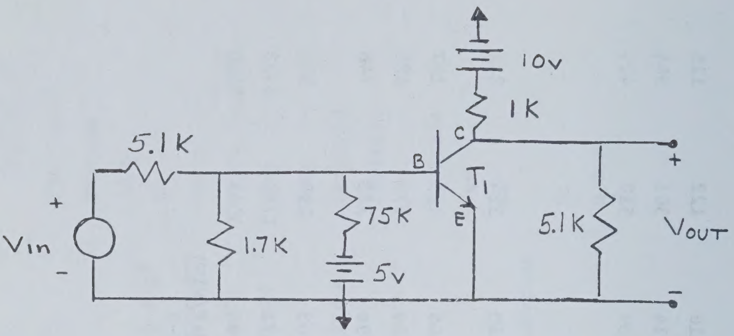

10v
1K
5.1K
B
C
T1
E
75K
1.7K
5v
5.1K
+
Vin
-
+
VOUT
-

Figure 1(a)

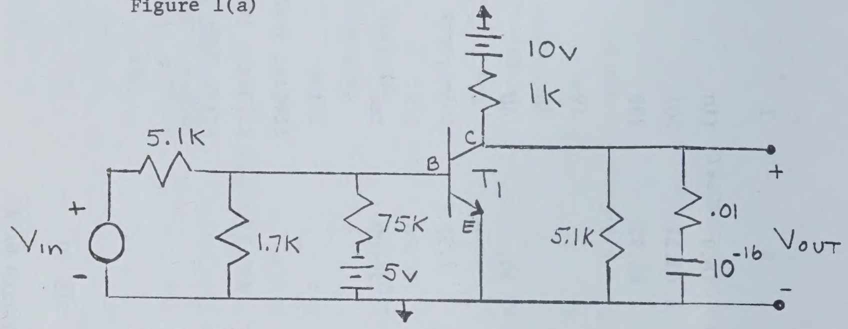
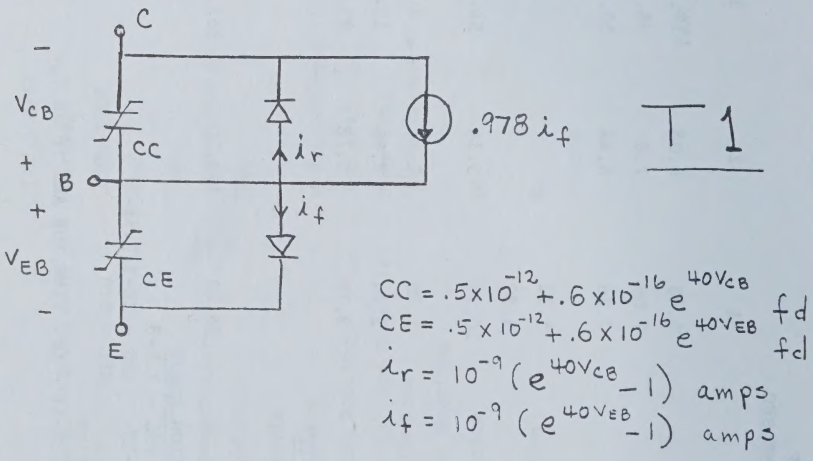

10v
1K
5.1K
B
C
T1
E
75K
1.7K
5v
5.1K
.01
10^{-16}
+
Vin
-
+
VOUT
-

Figure 1(b)


C
-
V_{CB}
+
B
+
V_{EB}
-
E
CC
CE
i_r
i_f
$.978\, i_f$
T 1

$$CC = .5\times 10^{-12} + .6\times 10^{-16} e^{40V_{CB}} \text{ fd}$$

$$CE = .5\times 10^{-12} + .6\times 10^{-16} e^{40V_{EB}} \text{ fd}$$

$$i_r = 10^{-9}\left(e^{40V_{CB}} - 1\right) \text{ amps}$$

$$i_f = 10^{-9}\left(e^{40V_{EB}} - 1\right) \text{ amps}$$

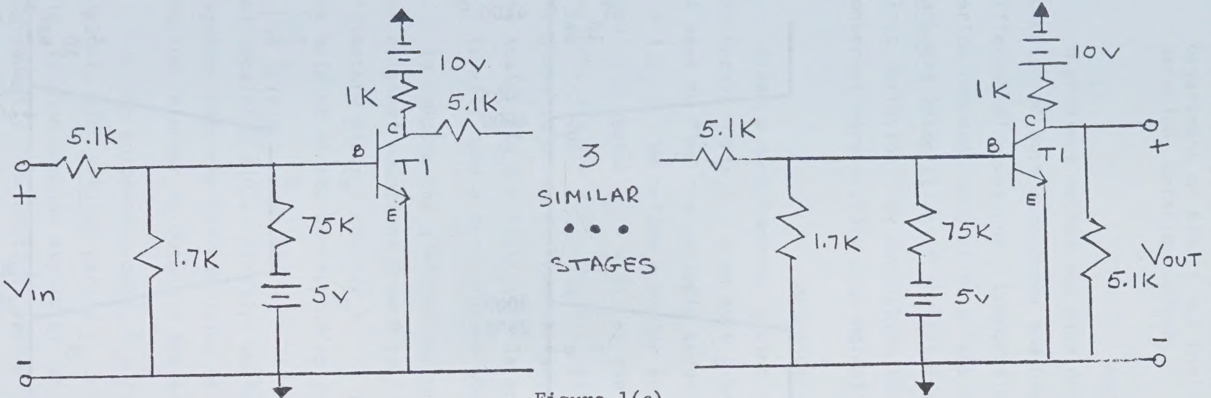

5.1K
+
Vin
-
1.7K
75K
5V
1K
10V
5.1K
B
C
E
T1
3
SIMILAR
STAGES
5.1K
1.7K
75K
5V
1K
10V
B
C
E
T1
+
VOUT
5.1K
-

Figure 1(c)

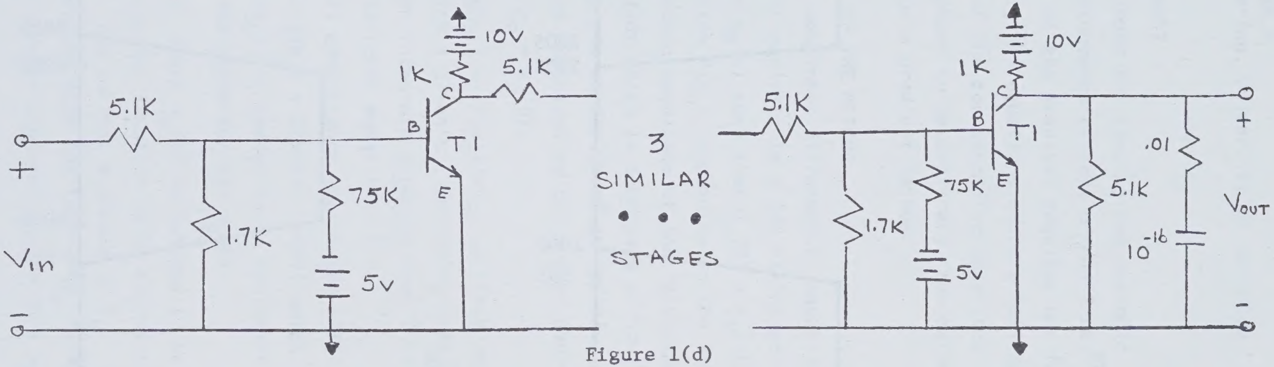

5.1K
+
Vin
-
1.7K
75K
5V
1K
10V
5.1K
B
C
E
T1
3
SIMILAR
STAGES
5.1K
1.7K
75K
5V
1K
10V
B
C
E
T1
5.1K
.01
10^{-16}
+
VOUT
-

Figure 1(d)

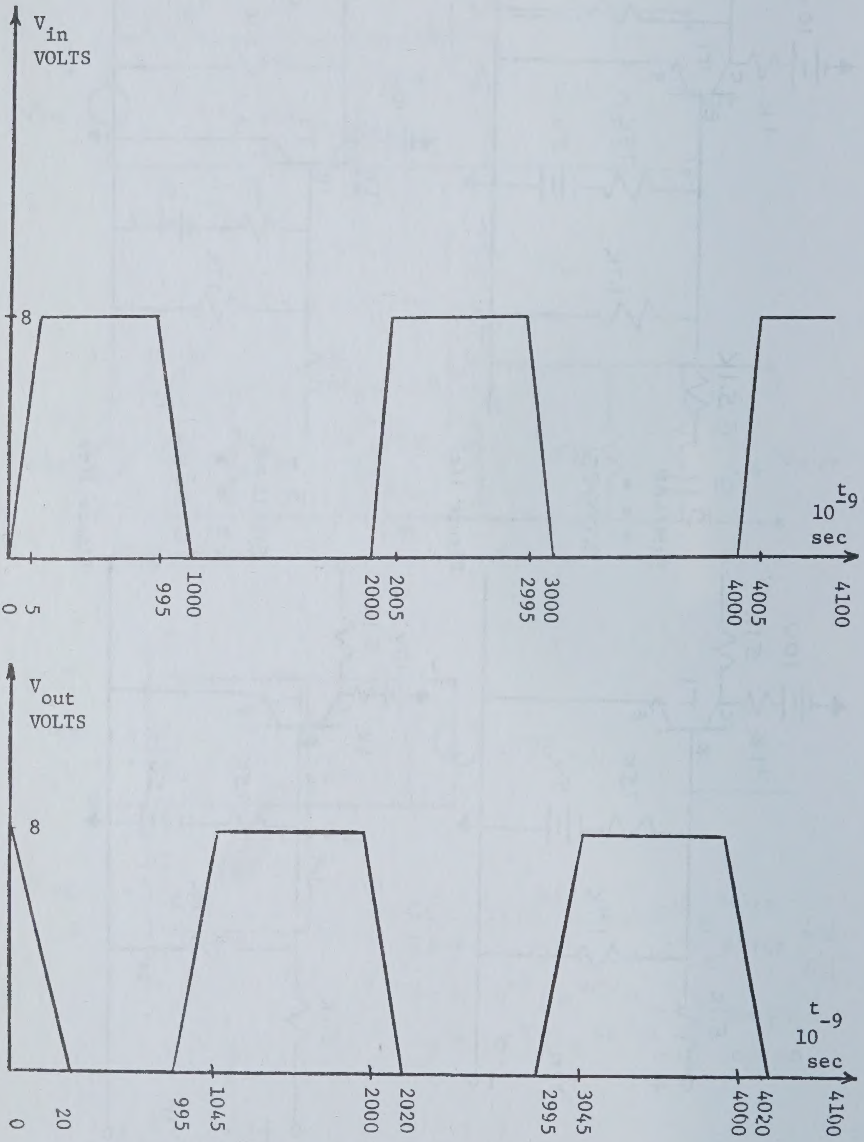

V in
VOLTS
8
t
10 -9
sec
0
5
995
1000
2000
2005
2995
3000
4000
4005
4100
V out
VOLTS
8
t
10 -9
sec
0
20
995
1045
2000
2020
2995
3045
4000
4020
4100

Figure 2

A GRADIENT METHOD FOR FINDING THE PERIODIC RESPONSE OF NONLINEAR SYSTEMS

M. S. NAKHLA AND F. H. BRANIN, JR.*
Department of Electrical Engineering, University of Waterloo,
Waterloo, Ontario, Canada

ABSTRACT

A gradient method has been developed for finding the periodic solution of both autonomous and nonautonomous systems of nonlinear differential equations. Computation of the gradient requires one full-period forward integration of the original equations and one full-period backward integration of the adjoint of the corresponding equations of first variation. By permitting the phase to be arbitrary, the method converges more rapidly than normal for a gradient method.

DESCRIPTION OF THE METHOD

Given a nonautonomous system of nonlinear differential equations of the form $\dot{\underset{\sim}{x}} = \underset{\sim}{f}(\underset{\sim}{x},t)$, with both $\underset{\sim}{x}$ and $\underset{\sim}{f}$ periodic in t and having period T, we seek to find the periodic solution $\underset{\sim}{x}_p(t)$ such that $\underset{\sim}{x}_p(0) = \underset{\sim}{x}_p(kT)$ for $k = 1,2,\ldots$ We define a scalar function $P(\underset{\sim}{x}_o)$, depending on the initial condition vector $\underset{\sim}{x}_o = \underset{\sim}{x}(0)$, as the square magnitude of the "discrepancy vector", $\underset{\sim}{\delta}(\underset{\sim}{x}_o) = \underset{\sim}{x}(T) - \underset{\sim}{x}_o = \int_0^T \underset{\sim}{f}(\underset{\sim}{x},\tau)d\tau$ which is a measure of how far any given solution is from satisfying the condition of periodicity. Thus the scalar $P(\underset{\sim}{x}_o) = ||\underset{\sim}{\delta}(\underset{\sim}{x}_o)||^2$ is to be minimized by the proper choice of $\underset{\sim}{x}_o$; it will have a zero minimum when $\underset{\sim}{x}_o = \underset{\sim}{x}_p(0)$.

To compute the gradient vector $\underset{\sim}{g}(\underset{\sim}{x}_o) = \partial P(\underset{\sim}{x}_o)/\partial \underset{\sim}{x}_o$, we first integrate the original equations from 0 to T with a given $\underset{\sim}{x}_o$, obtaining both the trajectory $\underset{\sim}{x}(t;\underset{\sim}{x}_o)$ and $\underset{\sim}{\delta}(\underset{\sim}{x}_o)$. We then integrate backward from T to 0 the adjoint of the corresponding variational equations, $\dot{\underset{\sim}{z}} = -\left[\frac{\partial \underset{\sim}{f}}{\partial \underset{\sim}{x}}\bigg|_{\underset{\sim}{x}(t;\underset{\sim}{x}_o)}\right]^t \underset{\sim}{z}$, using as "initial conditions" the vector $\underset{\sim}{z}(T) = \underset{\sim}{\delta}(\underset{\sim}{x}_o)$ and obtaining $\underset{\sim}{z}(0)$. Finally, we have $\underset{\sim}{g}(\underset{\sim}{x}_o) = 2[\underset{\sim}{z}(T) - \underset{\sim}{z}(0)]$ which, together with the current value of $P(\underset{\sim}{x}_o)$, is passed to an optimization routine in order to obtain a better approximation for $\underset{\sim}{x}_p(0)$.

In the autonomous case, $\dot{\underset{\sim}{x}} = \underset{\sim}{f}(\underset{\sim}{x})$, where $\underset{\sim}{x}_p(t)$ is assumed to be periodic with unknown period T_p, the scalar function to be minimized, $P(\underset{\sim}{x}_o,T)$, now depends explicitly on T, the current estimate of T_p. Consequently, the gradient vector involves both the term $\partial P(\underset{\sim}{x}_o,T)/\partial \underset{\sim}{x}_o$ and the term $\partial P(\underset{\sim}{x}_o,T)/\partial T$. The first term is computed in the same manner as in the nonautonomous case; the second term is found using the relation $\partial P(\underset{\sim}{x}_o,T)/\partial T = 2[\underset{\sim}{f}^t(\underset{\sim}{x}(T))\ \underset{\sim}{\delta}(\underset{\sim}{x}_o)]$. Again, the gradient vector, together with the current value of $P(\underset{\sim}{x}_o,T)$, is passed to an optimization routine in order to obtain better approximations to $\underset{\sim}{x}_p(0)$ <ins>and</ins> T_p.

---

* Visiting Professor, on special assignment from IBM Corporation, Kingston, N.Y.

DISCUSSION

This method has a wider region of convergence and proves to be more efficient in many cases than the Aprille-Trick method[1,2] which is based on a Newton-Raphson iteration procedure. In particular, the Aprille-Trick method requires for the autonomous case that one of the variables be assigned a fixed value at each iteration. This value, moreover, must lie within the (<ins>unknown</ins>) range of variation of the variable along the orbital trajectory describing the periodic solution; otherwise, the method will not converge. Our method is free of this restriction which, in effect, fixes the phase of the oscillation and forces the Aprille-Trick method to converge to a <ins>fixed</ins> point on the orbit. By allowing the phase to be arbitrarily determined by the iteration process, our method can converge on any point along the orbit and therefore has a much larger target to hit; consequently, convergence is found to be much faster than would normally be expected in a gradient method. We believe that this arbitrary phase approach can be extended to the nonautonomous case also and that it will speed convergence there, too; but we have not yet proved this in practice.

REFERENCES

1. T. J. Aprille and T. N. Trick, "Steady State Analysis of Nonlinear Circuits with Periodic Inputs", Proc. IEEE, Vol. 60, pp. 108-114, Jan. 1972.
2. T. J. Aprille and T. N. Trick, "A Computer Algorithm to Obtain the Steady State Response of Nonlinear Oscillators", IEEE Trans. Circuit Theory, Vol. CT-19, pp. 354-360, July 1972.

A NEW TECHNIQUE FOR PARTIAL FRACTION EXPANSION WITH MULTIPLE POLES

MUNETAKA KONOMI and NOBUYOSHI YOSHIDA
Transmission Division
Nippon Electric Company, Ltd.
Kawasaki, 211 JAPAN

ABSTRACT

A new simple and effective algorithm is derived for obtaining the coefficients in the partial fraction expansion of a rational function with multiple poles. The algorithm consists purely of algebraic operations and does not need any differential calculus. It can readily be adapted to a digital computer. Manual computation is also possible because of its simplicity.

INTRODUCTION

On the partial fraction expansion of a rational function, the traditional Heviside expansion theorem is well known. Application of this theorem to a function with multiple poles involves differential calculus. Therefore, this approach is tedious and is not easily adaptable to machine computation.

Several numerical techniques for the partial fraction expansion of a rational function with multiple poles without differential calculus have been reported [1]-[4].

Wehrhahn showed an algebraic method for this problem, avoiding differential calculus [1]. His method works in such a way that, starting with an auxiliary function with multiplicity of all the poles deleted, in each of the successive steps the order of the poles is augmented by one, until the proper multiplicity is reached. The effectiveness of his method is limited to the function whose numerator is a constant.

Karni and Fielder showed that Wehrhahn's method can be extended to proper functions [2], [3]. (A proper function is defined as being a function where the degree of the denominator is higher than that of the numerator.) On each step of their procedure, one has to find, by inspection, the terms that have been developed before.

The approaches above-mentioned are slow and time consuming, especially if the multiplicity of the pole is higher or if the function has more than one multiple pole.

Moad showed a different approach to the partial fraction expansion of a rational function with multiple poles [4]. Theoretically, his method is based on Heviside expansion theorem, but, in actual computation, differential calculus is not needed. From the machine computation point of view, his method seems to be one of the best approaches to the partial fraction expansion with multiple poles.

The objective of this report is to show another simple and effective procedure for obtaining the coefficients in the partial fraction expansion of a rational function with multiple poles. In the procedure presented here, the orders of poles and zeros are augmented one by one, until the proper multiplicities are reached. The partial fraction expansion of each step is directly and systematically obtained from that of the previous step. The procedure can be very easily adapted to a machine computation, because the algorithm needs neither inspection nor factorial computation.

PROBLEM DESCRIPTION

Let a rational function of s be

$$F(s) = H \frac{\sum_{j=1}^{m} (s-z_j)^{\beta_j}}{\sum_{i=1}^{n} (s-p_i)^{\alpha_i}} \qquad (1)$$

where H is a constant, n is the number of distinct poles, α_i is the multiplicity of pole p_i, m is the number of distinct zeros and β_j is the multiplicity of zero z_j. The problem is to find coefficients C_{ij} in the partial fraction expansion

$$F(s) = \sum_{i=1}^{n} \sum_{j=1}^{\alpha_i} \frac{C_{ij}}{(s-p_i)^j} \quad . \qquad (2)$$

Degrees δ and γ of the denominator and the numerator of Eq. (1) are $\delta = \sum_{i=1}^{n} \alpha_i$ and $\gamma = \sum_{j=1}^{m} \beta_j$, respectively, and it is assumed that $\delta > \gamma$.

PROCEDURES

Equation (1) may be written in the form

$$F(s) = H \frac{\sum_{\ell=1}^{\nu} (s - \eta_{\ell})}{\sum_{\mu=1}^{\delta} (s - \xi_{\mu})} \tag{3}$$

where $\xi_{\mu} = p_1$ (for $\mu : 1 \leqq \mu \leqq \alpha_1$),

$\xi_{\mu} = p_i$ (for $\mu : 1 + \sum_{\lambda=1}^{i-1} \alpha_{\lambda} \leqq \mu \leqq \sum_{\lambda=1}^{i} \alpha_{\lambda}$) (i=2, 3, ..., n),

$\eta_{\ell} = z_1$ (for $\ell : 1 \leqq \ell \leqq \beta_1$),

$\eta_{\ell} = z_j$ (for $\ell : 1 + \sum_{\lambda=1}^{j-1} \beta_{\lambda} \leqq \ell \leqq \sum_{\lambda=1}^{j} \beta_{\lambda}$) (j=2, 3, ...,m).

Function series $F^{(1)}(s)$, $F^{(2)}(s)$, ..., $F^{(\delta)}(s)$ is defined as follows.

$$\begin{cases} F^{(1)}(s) = \frac{H}{s-\xi_1} & (4\text{-}1) \\ F^{(k+1)}(s) = F^{(k)}(s) \frac{s-\eta_k}{s-\xi_{k+1}} \quad (1 \leqq k \leqq \nu) & (4\text{-}2) \\ F^{(k+1)}(s) = F^{(k)}(s) \frac{1}{s-\xi_{k+1}} \quad (\nu + 1 \leqq k \leqq \delta - 1) & (4\text{-}3) \end{cases}$$

The following shows a method in which, starting with $F^{(1)}(s)$, we proceed to $F^{(2)}(s)$, $F^{(3)}(s)$, ..., $F^{(\delta)}(s) = F(s)$.

Equation (4-2) can be rewritten as

$$F^{(k+1)}(s) = F^{(k)}(s) + (\xi_{k+1} - \eta_k) \underline{F^{(k)}(s) \frac{1}{s-\xi_{k+1}}} \; . \tag{5}$$

Since the underlined part of Eq. (5) has identical form with the right hand side of Eq. (4-3), the problem is reduced to the expansion of $F^{(k+1)}(s)$, which is in the form shown in Eq. (4-3).

Since $F^{(1)}(s)$ in Eq. (4-1) is already in partial fraction form, the problem at each step of the procedure is :

Given $F^{(k)}(s)$, in partial fraction form, as

$$F^{(k)}(s) = \sum_{i=1}^{n(k)} \sum_{j=1}^{\alpha_i^{(k)}} \frac{C_{ij}^{(k)}}{(s-p_i)^j} \tag{6}$$

find $C_{ij}^{(k+1)}$ of

$$F^{(k+1)}(s) = \sum_{i=1}^{n(k+1)} \sum_{j=1}^{\alpha_i^{(k+1)}} \frac{C_{ij}^{(k+1)}}{(s-p_i)^j} \tag{7}$$

where $n(k)$ is the number of distinct poles in $F^{(k)}(s)$ and $\alpha_i^{(k)}$ is the multiplicity of ith pole p_i in $F^{(k)}(s)$.

Substituting Eq. (6) into Eq. (4-3), we get

$$F^{(k+1)}(s) = \left\{ \sum_{i=1}^{n(k)} \sum_{j=1}^{\alpha_i^{(k)}} \frac{C_{ij}^{(k)}}{(s-p_i)^j} \right\} \frac{1}{s-\xi_{k+1}} = \sum_{i=1}^{n(k)} G_i^{(k+1)}(s) \tag{8}$$

where

$$G_i^{(k+1)}(s) = \left\{ \sum_{j=1}^{\alpha_i^{(k)}} \frac{C_{ij}^{(k)}}{(s-p_i)^j} \right\} \cdot \frac{1}{s-\xi_{k+1}} \quad .$$

ξ_{k+1} is either equal to $p_{n(k)}$ or distinct from any of the p_i values. The problem is to determine $n(k+1)$, $\alpha_i^{(k+1)}$ and $C_{ij}^{(k+1)}$ in Eq.(7).

Obviously

$$n(k+1) = \begin{cases} n(k) & (\xi_{k+1} = p_{n(k)}) \\ n(k)+1 & (\text{otherwise}) \end{cases} \tag{9}$$

In order to determine $\alpha_i^{(k+1)}$ and $C_{ij}^{(k+1)}$, each term $G_i^{(k+1)}(s)$ of Eq. (8) is expanded.

The following two cases are considered separately.

[Case I] Term $G_i^{(k+1)}(s)$ with an i value such that $p_i = \xi_{k+1}$ Such a term can be developed into

$$G_i^{(k+1)}(s) = \left\{ \sum_{i=1}^{\alpha_i^{(k)}} \frac{C_{ij}^{(k)}}{(s-p_i)^j} \right\} \cdot \frac{1}{s-\xi_{k+1}} = \sum_{j=2}^{\alpha_i^{(k+1)}} \frac{C_{ij}^{(k+1)}}{(s-p_i)^j} \tag{10}$$

and, obviously,

$$\alpha_i^{(k+1)} = \alpha_i^{(k)} + 1 \tag{11}$$

$$C_{ij}^{(k+1)} = C_{ij-1}^{(k)} \qquad (j=2,\ 3,\ \ldots,\ \alpha_i^{(k+1)}) \tag{12}$$

[Case II] Term $G_i^{(k+1)}(s)$ with an i value such that $p_i \neq \xi_{k+1}$
Such a term can be developed into

$$G_i^{(k+1)}(s) = \left\{ \sum_{i=1}^{\alpha_i^{(k)}} \frac{C_{ij}^{(k)}}{(s-p_i)^j} \right\} \cdot \frac{1}{s-\xi_{k+1}} = \sum_{j=1}^{\alpha_i^{(k+1)}} \frac{C_{ij}^{(k+1)}}{(s-p_i)^j} + \frac{h_i^{(k+1)}}{s-\xi_{k+1}} \tag{13}$$

and $\alpha_i^{(k+1)}$, $C_{ij}^{(k+1)}$ and $h_i^{(k+1)}$ are determined by the following procedure.

Procedure

(i) $\alpha_i^{(k+1)} = \alpha_i^{(k)}$

(ii) $W = \dfrac{1}{\xi_{k+1} - p_i}$

(iii) $C_{ij}^{(k+1)} = -C_{ij}^{(k)} W \quad (\text{for } j = \alpha_i^{(k+1)})$

(iv) $C_{ij}^{(k+1)} = (C_{ij+1}^{(k+1)} - C_{ij}^{(k)})W \quad (\text{for } j = \alpha_i^{(k+1)}-1,\ \alpha_i^{(k+1)}-2,\ \ldots,\ 2,\ 1,)$

(v) $h_i^{(k+1)} = -C_{i1}^{(k+1)}$

This procedure is derived from applying the equation

$$\frac{1}{(s-p_i)^j} \cdot \frac{1}{s-\xi_{k+1}} = \frac{W^j}{s-\xi_{k+1}} - \sum_{r=1}^{j} \frac{W^{j-r+1}}{(s-p_i)^r} \tag{14}$$

where $W = \dfrac{1}{\xi_{k+1} - p_i} \quad (p_i \neq \xi_{k+1})$

to each term of $G_i^{(k+1)}(s)$. Equation (14) is a special case of Eq. (6) in the Moad paper [4].

Now, all the $C_{ij}^{(k+1)}$ coefficients, except $C_{n(k+1),1}^{(k+1)}$, are determined. $C_{n(k+1),1}^{(k+1)}$ is calculated by summing up the second term coefficient $h_i^{(k+1)}$ of Eq. (13).

EXAMPLE

$$F(s) = 0.5 \frac{(s-2.5)\ (s-3.5)^2}{(s-1.5)^3 (s-0.5)^2 (s-1)^2} \tag{15}$$

is a function to be expanded. It can be rewritten in the form of Eq. (3).

$$F(s) = \frac{0.5\ (s-2.5)(s-3.5)(s-3.5)}{(s-1.5)(s-1.5)(s-1.5)(s-0.5)(s-0.5)(s-1)(s-1)} \quad . \tag{16}$$

$F^{(k)}(s)$ (k=1, 2, ⋯ ,7) are defined from Eqs. (4-1)-(4-3) and (5) as follows.

$$F^{(1)}(s) = \frac{0.5}{s-1.5} \tag{17-1}$$

$$F^{(2)}(S) = F^{(1)}(s)\ \frac{s-2.5}{s-1.5} = F^{(1)}(s)-F^{(1)}(s)\ \frac{1}{s-1.5} \tag{17-2}$$

$$F^{(3)}(s) = F^{(2)}(s)\ \frac{s-3.5}{s-1.5} = F^{(2)}(s)-2F^{(2)}(s)\ \frac{1}{s-1.5} \tag{17-3}$$

$$F^{(4)}(s) = F^{(3)}(s)\ \frac{s-3.5}{s-0.5} = F^{(3)}(s)-3F^{(3)}(s)\ \frac{1}{s-0.5} \tag{17-4}$$

$$F^{(5)}(s) = F^{(4)}(s)\ \frac{1}{s-0.5} \tag{17-5}$$

$$F^{(6)}(s) = F^{(5)}(s)\ \frac{1}{s-1} \tag{17-6}$$

$$F^{(7)}(s) = F^{(6)}(s)\ \frac{1}{s-1} \tag{17-7}$$

Each $F^{(k)}(s)$ is expanded in the partial fraction form successively. In order to show how easily the procedure works, the process of expanding $F^{(7)}(s)$, in the partial fraction form, from $F^{(6)}(s)$, which has already been expanded, is shown. The partial fraction expansion of $F^{(6)}(s)$ is

$$F^{(6)}(s) = \frac{-4}{(s-1.5)^3} + \frac{24}{(s-1.5)^2} + \frac{-81}{(s-1.5)} + \frac{-18}{(s-0.5)^2} + \frac{-69}{s-0.5} + \frac{150}{s-1} \quad . \tag{18}$$

From this and Eq. (17-7), all the coefficients of the partial fraction expansion of $F^{(7)}(s)$ are determined, as shown in Fig. 1. The arrows in the figure indicate the order of calculation.

CONCLUSION

The entire process consists purely of algebraic operations and needs neither differential nor factorial calculation. It can readily be adapted to a digital computer. Manual computation is also possible because of its simplicity.

ACKNOWLEDGMENT

The authors wish to express their thanks to Y. Ishizaki of the Nippon Electric Company, Ltd., for his helpful suggestions in preparing the manuscript.

REFERENCES

[1] E. Wehrhahn, "On partial fraction expansion with high-order poles," IEEE Trans. Circuit Theory, vol. CT-14, P.346, Sept. 1967.

[2] S. Karni, "Easy partial fraction expansion with multiple poles," Proc. IEEE, vol. 57, P.231, Feb. 1969.

[3] D.C. Fielder and S. Karni, "Supplementary remarks on'Easy partial fraction expansion with multiple poles'," Proc. IEEE, vol. 57, P.1769, Oct. 1969.

[4] M.F. Moad and S. Karni "On partial fraction expansion with multiple poles through derivatives" Proc. IEEE, vol. 57, P.2056, Nov. 1969.

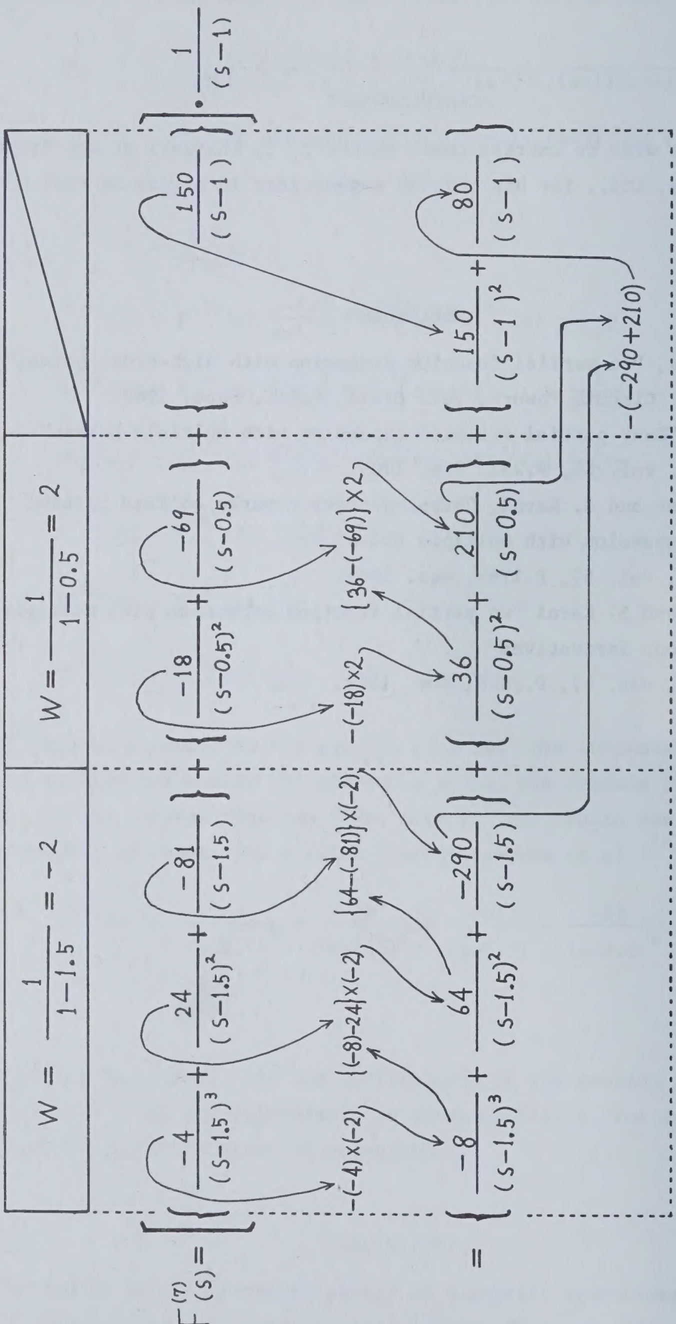

W = 1/(1−1.5) = −2
W = 1/(1−0.5) = 2
F(S)^(7) = [{ −4/(S−1.5)^3 + 24/(S−1.5)^2 + −81/(S−1.5) } + { −18/(S−0.5)^2 + −69/(S−0.5) } + { 150/(S−1) }] • 1/(S−1)
−(−4)×(−2)
{(−8)−24}×(−2)
{64−(−81)}×(−2)
−(−18)×2
{36−(−69)}×2
= { −8/(S−1.5)^3 + 64/(S−1.5)^2 + −290/(S−1.5) } + { 36/(S−0.5)^2 + 210/(S−0.5) } + { 150/(S−1)^2 + 80/(S−1) }
−(−290+210)

Fig. 1. An illustrative example

ABSOLUTELY STABLE INTEGRATION METHOD BASED ON THE LAPLACE TRANSFORM INVERSION

K. SINGHAL, J. VLACH, M. NAKHLA
University of Waterloo, Department of Electrical Engineering,
Waterloo, Ontario, Canada.

ABSTRACT

A method for the time domain solution of networks using complex arithmetic, is proposed. Its properties are markedly superior to the methods for numerical integration of differential equations using real arithmetic. In contrast to such methods, it is self-starting, explicit, one step, has low truncation error and is A-stable for any order of integration.

INTRODUCTION

It was shown in [1] that the numerical inversion of the Laplace transform can be done by using the formula

$$v(t) = -\frac{1}{t}\sum_{i=1}^{M} K_i\, V(z_i/t) \qquad (1)$$

where t is the time considered, V is a function of the complex variable s and z_i, K_i are the poles and residues, respectively, of

$$e^z \approx \xi_{M,N}(z) = \frac{\sum_{i=o}^{N}(M+N-i)!\binom{N}{i}z^i}{\sum_{i=o}^{M}(-1)^i(M+N-i)!\binom{M}{i}z^i} = \sum_{i=1}^{M}\frac{K_i}{z-z_i} \quad (\text{if } N < M) \qquad (2)$$

Formula (1) was derived from the Bromwich integral by first introducing a change of variable $z = st$ and then replacing e^z by its Padé approximation(2). We always select $N < M$.

It was shown that the method exactly inverts the first M+N+1 terms of the Taylor expansion of the exact response. In addition, for a proper choice of M,N, the Dirac Impulse (or its derivatives), appearing at the origin, are automatically and correctly taken into account by (1).

The property of matching the first portion of the Taylor series of the time response is essentially equivalent to all methods currently used for the numerical integration of differential equations. If one could reset the function V(s) after taking a "small" step and make another small step, the method would be exactly equivalent to the numerical integration of differential equations.

Such a resetting can be easily achieved when analyzing networks in the time domain. Each initial charge across any capacitor is replaced by an equivalent impulse current source in parallel with the capacitor. Similarly each initial current through an inductor can be represented as an impulse voltage source in series with the inductor (see table I of [1]). The resetting idea can be easily explained on the simple network shown in Figure 1. In the frequency domain, the initial condition is expressed by the current source across the capacitor. The nodal admittance matrix has the form

$$\begin{bmatrix} G_1 + sC & -sC \\ -sC & G_2+sC \end{bmatrix}\begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} J(s) \\ 0 \end{bmatrix} + \begin{bmatrix} CV_o \\ -CV_o \end{bmatrix} \qquad (3)$$

where V_1,V_2 are the voltages at the nodes 1,2. The current sources on the right hand side are written separately to indicate that the first one contains the independent source and the second one takes care of the initial condition. Notice that J(s) is generally some function of s.

Formula (1) and resetting is used in an algorithm as follows:

1. Select M,N and take z_i and K_i (fixed complex numbers) from tables
2. Select the step size $t = h$ and obtain z_i/h
3. Insert z_i/h for each s in (3) and solve for $V_{1,i}$, $V_{2,i}$. Repeat for all i, i=1,2...M.
4. Add weighted sums $V_{1,i}, V_{2,i}$ as required by (1), along with the division by h. This results in V_1, V_2 at time t=h.
5. Calculate $V_o = V_1 - V_2$, reset the value of the current source CV_o in (3) and go back to 3. for another step.

Since this method is equivalent to the integration of differential equations, we now establish the important properties of the method, namely its truncation error and its stability.

Truncation error

The truncation error is defined as the difference of the coefficient accompanying the first unmatched term of the Taylor expansion of the calculated response and that of the exact expansion. In our case, the error function is expressed by

$$\varepsilon(z) = e^{z} - \zeta_{M,N}(z) = \varepsilon_{M+N+1} z^{M+N+1} + \varepsilon_{M+N+2} z^{M+N+2} + \dots\dots \qquad (4)$$

Since the Padé approximation (2) is a rational function, its expansion can be formally written as

$$\frac{P_N(z)}{Q_M(z)} = \sum_{i=o}^{\infty} C_i z^i \qquad (5)$$

or

$$\sum_{i=o}^{N} a_i z^i = \left(\sum_{i=o}^{\infty} C_i z^i\right)\left(1 + \sum_{i=1}^{M} b_i z^i\right)$$

This expression can be rewritten in the form of an infinite matrix equation

$$\begin{bmatrix} a_o \\ a_1 \\ a_2 \\ \vdots \\ a_N \\ 0 \\ 0 \\ \\ \\ \end{bmatrix} = \begin{bmatrix} C_o & & & & \\ C_1 & C_o & & & \\ C_2 & C_1 & C_o & & \\ \vdots & & & & \\ C_N & C_{N-1} & C_{N-2} & \dots\dots\dots & C_o \\ & & & & \\ & & & & \\ & & & & \\ C_{M+N+1} & C_{M+N} & C_{M+N-1} & \dots\dots & C_{N+2} \end{bmatrix} \begin{bmatrix} 1 \\ b_1 \\ b_2 \\ \vdots \\ \\ b_M \\ 0 \\ 0 \\ \vdots \end{bmatrix} \qquad (6)$$

The two vectors in (6) are known, since the explicit formula is given by (2). The first M+N+1 coefficients C_i are also known, since we are approximating the function e^z and its Taylor expansion gives

$$C_i = 1/i\,! \qquad (7)$$

Thus the first unknown coefficient in (6) is C_{M+N+1} and can be found from the (M+N+2)-nd row of (6)

$$\sum_{i=o}^{M} C_{M+N+1-i}\, b_i = 0 \qquad (8)$$

Inserting for C_i from (7) and for b_i from the explicit formula (2) (normalized so that the constant term is unity) we obtain

$$C_{M+N+1} = -\frac{1}{(M+N)!} \sum_{i=1}^{M} \frac{(-1)^i}{(M+N+1-i)} \binom{M}{i} \tag{9}$$

Expression (9) gives the coefficient of the (M+N+2)-nd term of the Taylor expansion of $\xi_{M,N}(z)$ about $z = 0$ and includes the division by (M+N+1)!. Therefore, the derivative $D_{M,N}$ is equal

$$D_{M,N} = \frac{d^{M+N+1}}{dz^{M+N+1}} \xi_{N,M}(z) \Big|_{z=o} = (M+N+1)!\, C_{M+N+1} = (M+N+1) \sum_{i=1}^{M} \frac{(-1)^{i-1}}{(M+N+1-i)} \binom{M}{i} \tag{10}$$

Since, in addition,

$$\frac{d^{M+N+1}}{dz^{M+N+1}} e^z \Big|_{z=o} = 1, \tag{11}$$

the difference of the exact (11) and actual (10) value represents the truncation error

$$E_{M+N+1} = 1 - D_{M,N} \tag{12}$$

The values of (12) were calculated for various M,N and the following algorithm was found. Define

$$\alpha_{o,i} = \alpha_{i,o} = 1$$

Then $\qquad \alpha_{M,N} = \alpha_{M-1,N} + \alpha_{M,N-1}$

and $\qquad D_{M,N} = 1 - \dfrac{(-1)^M}{\alpha_{M,N}}$

The values of the truncation error for several low M,N are indicated in Table 1. It rapidly decreases for values of N not differing considerably from M. In all applications, as explained in [1], it is advisable to use N=M-1, M-2, M-3. Thus the method has a truncation error decreasing rapidly with increasing M,N.

The number of matched derivatives, equal to p = M+N, is called the order of integration.

Stability Properties

Stability properties of numerical methods for the integration of differential equations are always studied on the differential equation

$$y' = \lambda y \tag{13}$$

where λ may be either a real or a complex constant.

Introduce the Laplace transform for y'

$$L[y'] = sY - y_o$$

Then

$$Y = \frac{y_o}{s - \lambda}$$

To calculate the response at t = h, apply (1). V(s) is now $y_o/(s-\lambda)$. The result is

$$y_1 = y_o \left\{ \frac{-1}{h} \sum_{i=1}^{M} \frac{K_i}{z_{i/h} - \lambda} \right\} = y_o \left\{ \sum_{i=1}^{M} \frac{K_i}{\lambda h - z_i} \right\} \tag{14}$$

$$= y_o H$$

Introduce, for simplicity, a new variable

$$q = h\lambda \tag{15}$$

It can be seen that

$$H = \sum_{i=1}^{M} \frac{Ki}{q - z_i} = \xi_{M,N}(q) \tag{16}$$

is the approximation we used for the solution of the Bromwich integral. We can write

$$y_1 = H\, y_o$$

taking n steps, we have

$$y_n = H^n y_o$$

If $n \to \infty$, a stable solution is obtained for all values of q such that $|H| \leq 1$, i.e.,

$$|\xi_{M,N}(q)| \leq 1 \tag{17}$$

We first notice that

$$\xi_{M,N}(q) = \frac{1}{\xi_{N,M}(-q)} \tag{18}$$

and thus the stability boundaries (for the equal sign of (17)) for the cases $M > N$ will be the mirror images of those with $N > M$ about the vertical axes. In addition, due to the division in (18), the region outside the curves for $M > N$ will transform into that inside the curves for $M < N$.

The boundaries were established numerically by realizing that

$$|\xi_{M,N}(q)| = 1$$

is equivalent to writing

$$\frac{P_N(q)}{Q_M(q)} = e^{j\Psi}$$

where P_N, Q_M are the numerator and denominator, respectively, of (2).

If M=N, the boundary is the imaginary axis of the q plane. However, this case cannot be considered because the integrated function does not fulfil some requirements on the number of poles and zeros. This is discussed in detail in [1] or [2]. However, all cases with $M > N$ are valid. The regions of stability for M=N+1 are shown in Figure 2. All curves close in the right half plane and the unstable region is inside of the curves. The curves are tangent to the vertical axis at the origin and never cross into the left half plane for N=M -1 or N=M-2, M arbitrary. Analytical proof of this statement is also available.

The curve can be understood as follows: Consider (13) with Re $\lambda < 0$. This is a stable differential equation. Select a step size h and obtain the product $q = h\lambda$. If such a (generally complex) point lies in the stable region of the method, an error committed at any step during the integration does not propagate with the integration, for any number of steps. If such a property is true for all Re $\lambda < 0$, the method is called "absolutely stable" or A-stable. It is known that A-stability of the widely accepted linear multistep formulas (LMS) exists only for p = 1,2. Our method fulfills the condition on absolute stability for any order.

It is worth mentioning that the case M=1, N=0 results in $K_1 = -1$ and $z_1 = +1$. The stability region is equivalent to the inside of a unit circle with the centre at (1,0). Also, p = M+N = 1. This case is exactly equivalent to the well known Euler backward integration formula. For M=2, N=0, the poles and residues are complex, but only one function evaluation is necessary, since the poles and residues are complex conjugate. The stability region is still a closed curve in the right half plane (see Figure 2). This property cannot be achieved by any LMS formula.

Further properties of the method

As follows from the algorithm in section 1, no previous history is necessary to make a new step. Thus the method is a one-step method, not requiring any storage of previous solutions. No iteration is required if the network is linear. This is in marked contrast to the implicit LMS formulae which must iterate even for a linear network. Large step sizes are possible if p is selected large.

Comparison of the presented method and of the LMS formulae is given in Table 2.

CONCLUSION

A method was developed for the time domain solution of networks. A frequency analysis program can be used to calculate the time response. The method is one-step, explicit, can use arbitrary order of integration and is absolutely stable for any order.

REFERENCES

1. M. Makhla, K. Singhal, J. Vlach: Novel application of the numerical Laplace transform inversio. Proceedings, Annual Allerton Conference on Circuit and System Theory, University of Illinois, October 1973, pp. 661-667.

2. K. Singhal, J. Vlach: Computation of time domain response by numerical inversion of the Laplace transform. To appear in Journal of the Franklin Institute.

Table 1.

Truncation error for various values of M,N

M	N					
	0	1	2	3	4	5
0	1	1	1	1	1	1
1	-1	-1/2	-1/3	-1/4	-1/5	-1/6
2	1	1/3	1/6	1/10	1/15	1/21
3	-1	-1/4	-1/10	-1/20	-1/35	-1/56
4	1	1/5	1/15	1/35	1/70	1/126
5	-1	-1/6	-1/21	-1/56	-1/126	-1/252

Table 2.

Comparison of the properties of the proposed method and of the Linear Multistep Formulas

Property	LMS formula	Proposed method
Self starting?	no	yes
Order of integration?	6 - 10 (max)	46 (can be higher)
Absolutely stable?	not for p 2	yes for any p
Arithmetic ?	real	complex
Iteration for linear networks?	yes	no
Solution of nonlinear networks possible?	yes	yes

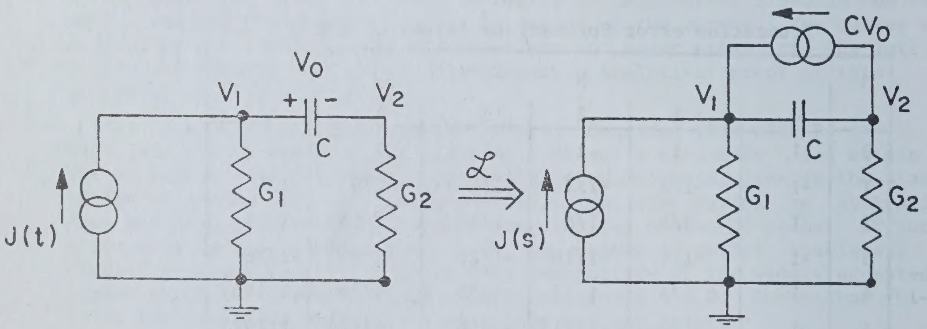
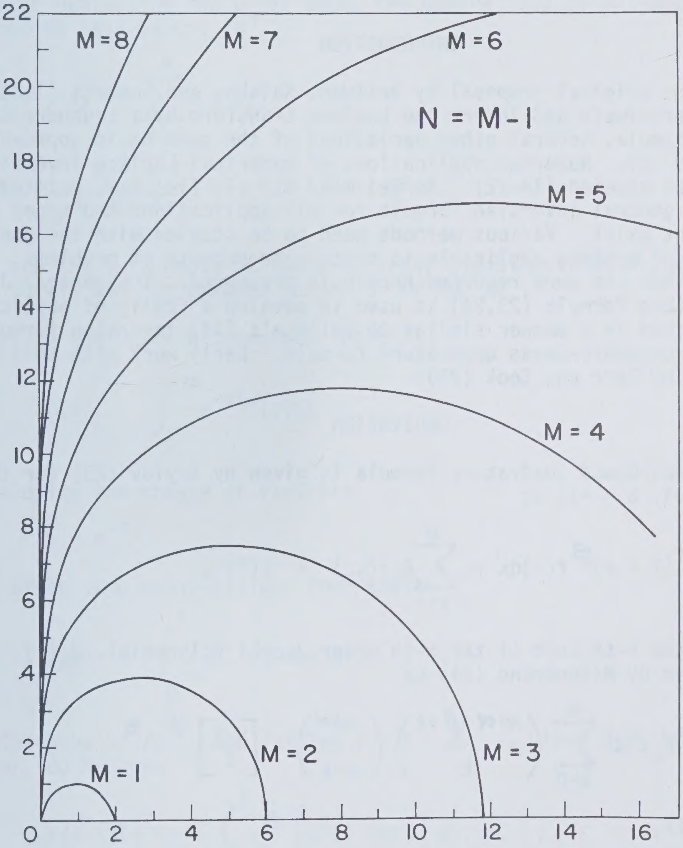

V_0
V_1
+
-
V_2
C
G_1
G_2
J(t)
$\mathcal{L}$
J(s)
CV_0
V_1
V_2
C
G_1
G_2

Figure 1.

Figure 2.

Stability boundaries for N = M-1. The stable region is inside of the curves. The curves are symmetric about the horizontal axis.



NUMERICAL LAPLACE INVERSION

GEORGE E. COOK and FRANCIS M. WELLS
Vanderbilt University, Nashville, Tennessee

ABSTRACT

The Jacobi-Gauss quadrature formula is used to develop a family of explicit inversion matrices for the Laplace transform in a manner similar to Bellman's inversion formula based on shifted Legendre polynomials. For different values of the parameter beta in the Jacobi polynomial, the mean square error is plotted versus a time scale factor for examples of increasing complexity.

INTRODUCTION

Since the original proposal by Bellman, Kalaba, and Lockett (1-8) of a method to approximate and invert the Laplace transform by a Legendre-Gauss quadrature formula, several other variations of the same basic approach have been tried (9-13). Numerous applications of numerical Laplace inversion have also been studied (14-22). As Bellman, et. al. (14) have pointed out, a completely general inversion formula for all applications and types of problems does not exist. Various methods need to be studied with the goal of finding sets of methods applicable to particular subsets of problems. It is to this end that the work reported herein is presented. The general Jacobi-Gauss quadrature formula (23,24) is used to develop a family of explicit inversion matrices in a manner similar to Bellman's (14) inversion formula based on the Legendre-Gauss quadrature formula. Early work with this method was reported by Carr and Cook (25).

DERIVATION

The Jacobi-Gauss quadrature formula is given by Krylov (23) for the interval (a = -1, b = +1) as

$$\int_{-1}^{1} (1 - x)^{\alpha} (1 + x)^{\beta} f(x)dx = \sum_{k=1}^{m} A_k f(x_k) + E(f) , \qquad (1)$$

where x_k is the k-th zero of the m-th order Jacobi polynomial, $J_m(x)$. $J_m(x)$ is given by Hildebrand (24) as

$$\emptyset_m(x) = (-1)^m C_m 2^m m! \sum_{k=0}^{m} \binom{m+\alpha+\beta+k}{k} \binom{m+\alpha}{m-k} \left[\frac{x-1}{2}\right]^k \qquad (2)$$

$$= C_m J_m(x),$$

where $C_m = (-1)^m/2^m m!$

Since the quadrature formula is to be used to approximate the Laplace transform integral, it is convenient to use the change of variable (3), on equation (1) to change the interval from (-1,+1) to (0,+1). The quadrature

$$\begin{aligned} x &= 2y - 1 \qquad & x = -1,\ y = 0 \\ dx &= 2dy \qquad & x = +1,\ y = +1 \end{aligned} \qquad (3)$$

formula is now given by

$$2^{\alpha+\beta+1}\int_0^1 (1-y)^{\alpha} y^{\beta} f(y)dy = \sum_{k=1}^{m} A_k f(y_k) + E(f). \tag{4}$$

The y_k are roots of a new polynomial, $P_m(y)$, which is orthogonal over the interval (0,+1) and is obtained by applying the previous change of variable to the old Jacobi polynomial, Jm(x). The weighting on each of the sample values in equation (4) remains unchanged except for the effect of the change of variable $x = 2y - 1$. For consideration here the value of alpha is assumed equal to zero and the $2^{\alpha+\beta+1}$ factor is lumped with the weighting values, Ak. After neglecting the error term, the Generalized Jacobi-Gauss Quadrature, as termed in this paper, is

$$\int_0^1 y^{\beta} f(y)dy \cong \sum_{k=1}^{m} H_k f(y_k), \tag{5}$$

where $H_k = \dfrac{A_k}{2^{\alpha+\beta+1}}$, $k = 1,2,\ldots,m$,

and y_k are the k-th roots of the m-th order Jacobi polynomial, $P_m(y)$, defined over (0,+1) and for assumed values of α and β .

Taking the Laplace transform integral,

$$F(s) = \int_0^{\infty} e^{-st} f(t)dt \tag{6}$$

and employing the change of variable

$$x = e^{-t} \tag{7}$$

gives, after some manipulation, the results

$$F(s) = \int_0^1 x^{s-1} f(-\ln x)dx. \tag{8}$$

Multiplying by x^{β}/x^{β} and letting $k = s - \beta - 1$, where k is an integer, equation (8) becomes

$$F(k+\beta+1) = \int_0^1 x^{\beta}\left[x^k f(-\ln x)\right]dx. \tag{9}$$

The Generalized Jacobi-Gauss quadrature formula, given by equation (5), can be used to approximate equation (9) and gives

$$F(k+\beta+1) = \frac{1}{2^{\beta+1}} \sum_{i=1}^{m} H_i x_i^{\,k} f(-\ln x_i), \qquad k = 0,1,2,\ldots,m-1, \tag{10}$$

where x_i, $(i = 1,2,\ldots,m)$, is the i-th zero of the m-th order Jacobi polynomial, and where an assumed value of beta has been chosen. Since the values in the matrix, $H_i x_i^k$, do not change for different sets of sampled values, the matrix can be generated once and for all and stored in computer memory for later use. It is only necessary to choose parameters β and m.

Explicit Inversion

The task of solving equation (10) or to generate an inverse for the Laplace transform approximation is not a straightforward matter as has been pointed out by Bellman. Since arbitrary small changes in F(s) may cause large changes in f(t), one describes the matrix in (10) as "ill-conditioned".

In order to take advantage of the special analytic form of equation (10) to generate an inverse, a few preliminary modifications must be carried out. Multiplying the k-th equation in (10) by q_k, as yet unspecified, and adding the corresponding terms in all of the equations gives

$$\sum_{k=0}^{m-1} q_k F(k+\beta+1) = \sum_{k=0}^{m-1} q_k \frac{1}{2^{\beta+1}} \sum_{i=1}^{m} H_i y_i^k f(y_i)$$

$$= \frac{1}{2^{\beta+1}} \sum_{i=1}^{m} H_i f(y_i) \sum_{k=0}^{m-1} q_k y_i^k, \tag{11}$$

where $y_i = -\ln x_i$.

Now let

$$s(y) = \sum_{k=0}^{m-1} q_k y^k \tag{12}$$

and equation (11) can be rewritten as

$$\sum_{k=0}^{m-1} F(k+\beta+1) q_k = \frac{1}{2^{\beta+1}} \sum_{i=1}^{m} H_i f(y_i) s(y_i) \tag{13}$$

where $s(y_i)$ is a polynomial of degree m-1 chosen in some arbitrary manner for convenience.

By using conditions of orthogonality where

$$s_j(y_i) = 0,\ i \neq j \qquad s_j(y_i) = 1,\ i = j, \tag{14}$$

equation (12) can be expressed as

$$s_j(y) = \sum_{k=0}^{m-1} q_{kj} y^k. \tag{15}$$

Equation (13) can be reduced and expressed as

$$f(y_j) = 2^{\beta+1} \sum_{k=0}^{m-1} F(k+\beta+1) \left[q_{kj}/H_j\right] \qquad (16)$$

where the q_{kj} are determined by the conditions given in (14).

Using the Lagrange interpolation formula, equation (15) can be written as

$$s_j(y) = \sum_{k=0}^{m-1} q_{kj}{}^{k} = \frac{J_m(y)}{(y-y_j)J_m{}'(y_j)} \quad . \qquad (17)$$

For purposes of digital computer computation, the above equation is easily expanded in terms of the proper Jacobi polynomial and its evaluated derivative. The desired q_{kj} are the coefficients in the polynomial of degree m-1 on the left side of the equation. By repeating the expansion for each value of j, one obtains the desired inverse matrix

$$\left[q_{kj}/H_j\right] \quad . \qquad (18)$$

In order to implement the inverse matrix, the values obtained from the explicit inversion procedure can be computed (for specific values of beta and m) once and for all and tabulated. As shown by equation (16), multiplication of the inverse matrix by the appropriate values of the s-domain function and summation over k yields the approximate values of f(t) at the points $t = -\ln x_j$, $j = 1,2,\ldots,m$.

QUADRATURE DATA

Algorithm Used

Krylov (23) uses the earlier work of Szego (26) to show that the weights can be expressed as

$$A_k = \frac{2^{\alpha+\beta+1}\Gamma(\alpha+m+1)\Gamma(\beta+m+1)}{m!\,\Gamma(\alpha+\beta+m+1)(1-x_k{}^2)\left[P_m{}'(x_k)\right]^2} \quad . \qquad (19)$$

As stated previously, the desired polynomial employed in this paper to generate quadrature formulae must be orthogonal over the interval (0,1) and is obtained from $P_m(x)$ by a simple change of variable

$$x = 2y - 1. \qquad (20)$$

This substitution yields a new polynomial, $Q_m(y)$, which is orthogonal over (0,1) and has zeros, y_k, $k = 1,2,\ldots,m$. Employing the change of variable (20) in (19) gives the new weighting values for the quadrature over (0,1) as

$$H_k = \frac{2^{\alpha+\beta+1}\Gamma(\alpha+m+1)\Gamma(\beta+m+1)}{m!\,\Gamma(\alpha+\beta+m+1)(1-y_k)4y_k\left[Q_m{}'(y_k)\right]^2} \quad . \qquad (21)$$

The actual form of (21) computed had the factor $2^{\alpha+\beta+1}$ deleted such that agreement was made with the final form of the Jacobi-Gauss quadrature, given in equation (5).

Computer Techniques

Computations were carried out at the Vanderbilt University Computer Center on a Sigma 7 computer using double precision arithmetic and the FORTRAN IV programming language. The general procedure followed in the programming was to use a main routine to provide, in response to a particular choice of parameter beta, for the generation of an appropriate polynomial in terms of its coefficients. After obtaining a specific polynomial, a subroutine called POLRT is used to factor the polynomial and provide roots. POLRT is an IBM package subroutine described in reference (27) which computes the real and complex roots of a real polynomial. A search procedure is employed wherein incremental values are determined from a Newton-Raphson method and iteration is stopped after incremental changes are less than some preassigned limit. Error indications are given by the subroutine to show lack of convergence after 500 iterations on five different starting values. The final iterations on each root are performed on the original polynomial instead of a reduced form so that accumulated errors will be avoided. Double precision arithmetic was again used to provide highest accuracy obtainable with the computing facilities available.

The main routine also generates the weighting values from the previously described algorithm, and the data are used to compute the forward matrix for the Laplace transform and its explicit inverse. Checks are made to see if the forward and inverse matrices can be multiplied to give a unity matrix, and thus a check is performed upon the accuracy of the explicit inversion.

Comparison With Bellman's Data

Since the accuracy of these quadrature methods is greatly affected by the accuracy of the roots and weights that have been computed, a comparison is made with a typical set of data used by Bellman in his shifted Legendre method. The shifted Legendre is the same quadrature as the method herein where beta = 0. Table 1 gives the results of the Jacobi method for seventh order and compares the data with that obtained and used by Bellman (14). The results of Table 1 show agreement to about thirteen-place accuracy.

TABLE 1

COMPARISON OF DATA OBTAINED BY JACOBI METHOD
WITH SHIFTED LEGENDRE METHOD, m = 7

Roots

Jacobi ($\beta = 0$)	Shifted Legendre
0.025446043828261	0.025446043828261
0.129234407200303	0.129234407200303
0.297077424311301	0.297077424311301
0.499999999999994	0.500000000000000
0.702922575688687	0.702922575688698
0.870765592799816	0.870765592799697
0.974553956171375	0.974553956171379

Weights

Jacobi ($\beta = 0$)	Shifted Legendre
0.064742483084513	0.064742483084435
0.139852695744623	0.139852695744638
0.190915025252482	0.190915025252559
0.208979591836839	0.208979591836735
0.190915025252521	0.190915025252559
0.139852695744615	0.139852695744638
0.064742483084432	0.064742483084435

CHOICE OF BETA

The implementation of the quadrature method begins with a choice being made as to the value of the parameter beta that is to be used in a particular example. This choice affects the quadrature data that is to be used, and it also may allow one to obtain greater accuracy in the approximation by more optimum spacing of the sample points. To illustrate the choice available with this Jacobi-Gauss method, sets of quadrature data are given for the seventh order case and all values of beta between -1 and +1. Figure 1 gives the polynomial roots, and the corresponding sampling times are shown in Figure 2. Weighting values are found in Figure 3. To obtain a set of data from these curves, one should choose a value of beta and extend a vertical line through the family of curves. The intersection of the curves with the line gives the set of values associated with that figure. Of course, values obtained from the curves are not accurate enough to use in a real example, but a general picture of the influence of beta upon the data is formed. All roots are located between 0 and 1 and do not appear to be widely distributed over the range of beta given in the figures. Taking the negative logarithm of the roots gives the wide spread to the sample points as shown in Figure 2. Times nearest the origin have little change while those farthest from the origin have an almost exponential distribution. The distribution of the weights is both interesting and important. For larger values of beta all weithts are small, and for the opposite extreme of beta values the weights diverge and weights No. 1 and No. 2 are indeed quite large. For beta equal to zero, the weight curves cross each other.

Some limiting values of beta are evident and no consideration of values outside this useful range will be considered here. For beta less than -1.0, the roots are negative and the natural logarithm is undefined and no sampling times are defined. Values of beta greater than 1.0 give sampling times which are all near each other and are weighted by small numbers. This data is usable but not desirable from the standpoint of sampling over a wide range of the time function. Thus, consideration is only given those values of beta in the range $-1 < \beta < +1$.

TIME SCALING

A useful technique that has been employed in other forms of quadrature approximations of the Laplace transform is that of time scaling. The time scaling property of the Laplace transform is

$$\mathcal{L}\left[f(t/t_0)\right] = \int_0^\infty e^{-st} f(t/t_0)\,dt \quad , \tag{22}$$

where t_0 is the scaling constant.
Using this time scaling property, equation (10) becomes

$$t_0 F\left[t_0(k+\beta+1)\right] = \frac{1}{2^{\beta+1}} \sum_{i=1}^{m} H_i x_i^{k} f(-\ln x_i/t_0) \tag{23}$$

and thus the explicit inverse is

$$f(-\ln x_i/t_0) = \sum_{k=0}^{m-1} t_0 F\left[t_0(k+\beta+1)\right]\left[q_{ki}/H_i\right] \quad , \qquad i = 1,2,\ldots,m. \tag{24}$$

Figure 1. Distribution of Roots for Values of Beta

Note: Integers show correlation between a root and its sampling time and weight in Figures 2 and 3

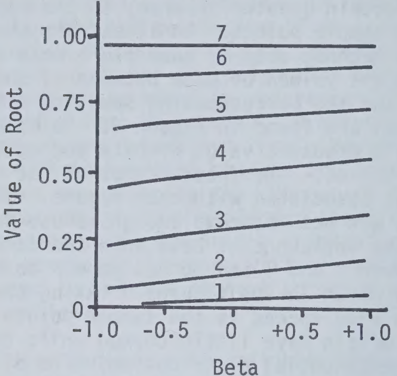


Figure 2. Distribution of Sampling Times for Values of Beta, m = 7

Note: Integers show correlation between a sample time and its root and weight in Figures 1 and 3

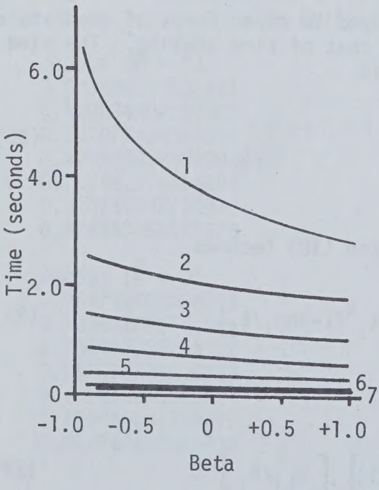
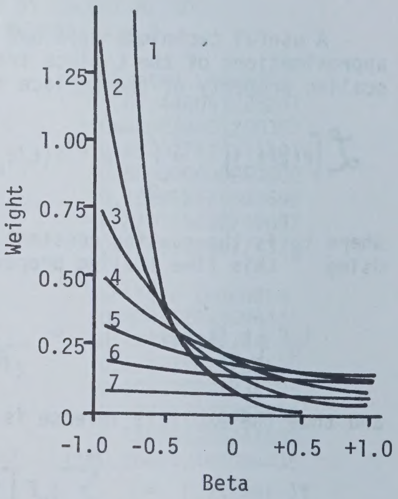


Figure 3. Distribution of Weights for Values of Beta, m = 7

Note: Integers show correlation between a weight and its root and sample time in Figures 1 and 2



This time scaline property is important in that it allows a linear shifting of the sampling times in the quadrature approximation. The forward matrix, $H_i x_i^k$, and the inverse matrix, q_{ki}/H_i, are not changed by the scaling procedure. The matrix data can be computed once and for all and tabulated (or stored in computer memory) and later used in many different problems with different scaling constants. The effect of the time scaling operation is to linearly displace the values of sample times for a particular set of quadrature data.

It should be pointed out that the time scaling operation is not the equivalent of using a different value of beta and its associated polynomial roots. As previously shown, the result of using different values of beta is to vary the spread of the samples which are farthest from the origin and to do so in a logarithmic sense. Time scaling provides a purely linearly displacement of all samples in either direction along the time axis and for a much wider range than is possible with that of varying beta.

INVERSION EXAMPLE

Consider the damped sinusoidal step response

$$f(t) = 1.0 - \frac{\exp(-\zeta w_n t)}{(1-\zeta^2)^{1/2}} \sin\left[w_n(1 - \zeta^2)^{1/2} t + \cos^{-1}\zeta\right] \qquad (25)$$

where $\zeta = 0.5$ and $w_n = 2.0$. The Laplace transform is given by

$$F(s) = w_n^2 / \left[s(s^2 + 2\zeta w_n s + w_n^2) \right] . \qquad (26)$$

The explicit inverse matrix (24) was used along with the exact values of the s-domain data to obtain the results given in Table 2. The time scale factor was $t_o = 1/7$, and beta was -1/2. Ten samples were utilized in the approximation. The data in Table 2 have been rounded off to five digits.

TABLE 2
INVERSION EXAMPLE

m = 10 Beta = -1/2 $t_o = 1/7$

Time (seconds)	Exact f(t)	Approximate f(t)	Percent Error
0.09653	0.01744	0.01782	+2.19931
0.51370	0.35497	0.35497	+0.00104
1.28601	1.04233	1.04077	-0.14994
2.45567	1.08243	1.08716	+0.43650
4.09619	0.98158	0.97318	-0.85550
6.33461	1.00107	1.00975	+0.86784
9.40304	1.00009	0.99486	-0.52360
13.77100	1.00000	1.00298	+0.29814
20.71089	1.00000	0.99841	-0.15902
35.98165	1.00000	1.00050	+0.05020

CONCLUSIONS

The method presented herein has been used by the authors to compute both the approximate Laplace transform, equation (10), and the explicit inverse, equation (16), for a number of examples of increasing complexity. It has been found that for functions consisting of sums of exponentials, the mean square error is minimum at time scale factors approximately corresponding to integer multiples of the dominant time constant. The accuracy of the method has been found to be equivalent or comparable to the technique of Bellman. Plots of the mse are shown in the oral presentation.

REFERENCES

1. Bellman, R., Kalaba, R., and Lockett, J., "Numerical Solution of Functional Equations by Means of Laplace Transforms - I: Renewal Equation", The Rand Corporation RM-3948-NIH, December, 1963.
2. Bellman, R., Kalaba, R., and Lockett, J., "Numerical Solution of Functional Equations by Means of Laplace Transforms - II: Differential-Difference Equations", The Rand Corporation, RM-3952-NIH, December, 1963.
3. Bellman, R., Kalaba, R., and Lockett, J., "Numerical Solution of Functional Equations by Means of Laplace Transforms - III: The Diffusion Equation", The Rand Corporation, RM-3974-NIH, January, 1964.
4. Bellman, R., Kalaba, R., and Lockett, J., "Numerical Solution of Functional Equations by Means of Laplace Transforms - IV: Nonlinear Equations", The Rand Corporation, RM-4088-NIH, May, 1964.
5. Bellman, R., Kalaba, R., and Lockett, J., "Numerical Solution of Functional Equations by Means of Laplace Transforms - V: Mismatched Systems of Linear Differential Equations", The Rand Corporation, RM-4089-NIH, May, 1964.
6. Bellman, R., Kalaba, R., and Lockett, J., "Numerical Solution of Functional Equations by Means of Laplace Transforms - VI: Stochastic Time-Optimal Control", The Rand Corporation, RM-4119-PR, May, 1964.
7. Bellman, R., "Numerical Solution of Functional Equations by Means of Laplace Transforms - VII: The Mellin Transform", The Rand Corporation RM-4179-NIH, June, 1964.
8. Bellman, R., Kalaba, R., and Lockett, J., "Numerical Solution of Functional Equations by Means of Laplace Transforms - VIII: Determination of Weighting Functions", The Rand Corporation, RM-4213-NIH, September, 1964.
9. Cook, G., Denman, E., and Carr, H., "Numerical Inversion of Laplace Transforms by the Laguerre-Gauss Quadrature Approximation", IEEE Trans. Automatic Control, Vol. AC-12, October, 1967.
10. Felts, W., and Cook, G., "On the Numerical Approximation of the Laplace Transform", IEEE Trans. Automatic Control, Vol. AC-14, June, 1969.
11. Carr, H., and Cook, G., "An Improved Multiple Interval Sampling Technique for Use With Gaussian Quadrature", IEEE Trans. Automatic Control, Vol. AC-15, No. 2, April, 1970.
12. Piessens, R., "Numerical Inversion of The Laplace Transform", IEEE Trans. Automatic Control, Vol. AC-14, June, 1969.
13. Schoenberg, I. J., "Remarks Concerning a Numerical Inversion of the Laplace Transform Due to Bellman, Kalaba, and Lockett", Journal of Mathematical Analysis and Applications, Vol. 43, pp. 823-828, 1973.
14. Bellman, R., Kalaba, R., and Lockett, J., Numerical Inversion of The Laplace Transform, American Elsevier Publishing Company, Inc., 1966.
15. Felts, W., "Techniques of System Identification Using Legendre-Gauss Quadrature", Master's Thesis, Vanderbilt University, January, 1969.
16. Carr, H., and Cook, G., "Application of the Laguerre-Gauss Quadrature Approximation to a Class of System Identification Problems", Record of the IEEE 1968 Region III Convention, pp. 7.4.1-7.4.4, April, 1968.
17. Cook, G., Carr, H., and Fernandes, R., "A Transfer Function Approximation Technique", Proceedings of 1972 Southeastern Symposium On Systems Theory, pp. 9-15, April, 1972.
18. Bellman, R., Kagiwada, H., and Kalaba, R., "Identification of Linear Systems Via Numerical Inversion of Laplace Transforms", IEEE Trans.

Automatic Control, Vol. AC-10, pp. 111-112, January, 1965.

19. Carr, H. M., "Application of the Laguerre-Gauss Quadrature to a Class of System Identification Problems", M.S. Thesis, Electrical Engineering Dept., Vanderbilt University, August, 1967.
20. Downs, S. W., "System Identification by Numerical Inversion of the Laplace Transform", M.S. Thesis, Electrical Engineering Dept., Vanderbilt University, August, 1967.
21. Pujol, A., "Numerical Inversion of Laplace Transforms Applied to Classical Control Theory", M.S. Thesis, Electrical Engineering Dept., Vanderbilt University, January, 1970.
22. Pujol, A., and Cook, G., "A Numerical Time Domain Solution of Linear Closed Loop Systems", Proceedings of the Ninth Annual IEEE Region III Convention, April, 1971.
23. Krylov, V. I., Approximate Calculation of Integrals, The MacMillan Co., New York, 1962.
24. Hildebrand, F. B., Introduction to Numerical Analysis, McGraw-Hill Book Co., Inc., New York, 1956.
25. Carr, H., and Cook, G., "Explicit Inversion of the Laplace Transform by a Jacobi-Gauss Quadrature Approximation", IEEE Trans. Automatic Control, Vol. AC-14, No. 6, December, 1969.
26. Szego, G., "Orthogonal Polynomials", American Mathematical Society Colloquium Publications, Vol. XIII, rev. ed., American Math. Soc., New York, 1959.
27. System/360 6ientific Subroutine Package (360A-CM-03X) Version III, Programmer's Manual H20-0205-3, IBM Technical Publications Dept., White Plains, N. Y.

ACCURATE COMPUTATION OF THE TRANSIENT RESPONSE WITH TIGHT ERROR BOUNDS

R. M. Wadhwani and R. M. Kline
Electrical Engineering Department
Washington University
St. Louis, Missouri 63130

N. B. Rabbat
IBM Corporation
Computer Aided Design Department
Hopewell Junction, New York 12533

ABSTRACT

This paper is concerned with the accurate computation of the transient response. Three major aspects of the new techniques and derivations are: (1) the criterion for choosing the number of terms in $\exp(\bar{A}t)$ matrix series; (2) the error bound if $\exp(\bar{A}t)$ matrix is truncated after M terms; (3) the basic limitations and range of applicability of the method. The savings in speed and computer storage are illustrated.

1. INTRODUCTION

In recent papers, Muraleedharan and Stromsmoe [5], Valant [4], and Thompson [3] have discussed extensions to Liou's [1,2] method of evaluating the transient response of a linear network whose transfer function is represented by the ratio of two polynomials in Laplace domain, i.e.

$$Y(s) = \frac{C_n s^{n-1} + C_{n-1} s^{n-2} + \dots C_1 s^0}{s^n + a_1 s^{n-1} + \dots + a_n} \quad . \tag{1}$$

The general solution of (1) at any time t is given by [1]

$$\underline{Y}(t) = \exp(\bar{A}t)\underline{Y}(0+) \tag{2}$$

where

$$\underline{Y}(t) = [y(t), y'(t), y''(t), \dots y^{n-1}(t)]^T \tag{3}$$

is a state vector with n state variables; the first state is the transient response; and

$$\bar{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & \dots & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & 0 & \dots & 1 \\ -a_n & -a_{n-1} & -a_{n-2} & -a_{n-3} & \dots & -a_1 \end{bmatrix} \tag{4}$$

Liou's method [1,2] yields the solution for all the state variables given the state vector at time t = 0. But, if one is only interested in the transient response, then it is not necessary to compute the rest of the states. Based on the above argument the general expression for y(t) is [4,5]

$$y(t) = \underline{C}\ \exp(\bar{A}t)\underline{B} \tag{5}$$

where

$$\underline{C} = [C_1, C_2, C_3, \ \ldots\ C_n]$$

and

$$\underline{B} = [0, 0, \ldots\ldots, 1]^T$$

If the infinite series representing $\exp(\bar{A}t)$ is truncated after M terms, then it can be easily shown that y(t) in (5) reduces to the following:

$$y(t) = \underline{C}\bar{\Phi}\underline{\Psi} \tag{6}$$

where $\bar{\Phi}$ is an n × M matrix and

$$\underline{\Psi} = {}_t[1, t, t^2/2!, \ldots, t^{M-1}/(M-1)!] \tag{7}$$

It can be shown analytically that the jth column of the Φ matrix is the last column of the $\bar{A}^{j-1}$ matrix, and the first column is the last column of the identity matrix.

It has been pointed out that only time dependent term in (6) is the column vector, $\underline{\Psi}$, because $\underline{C}\bar{\Phi}$ product will be a constant row vector, which for a given system is evaluated only once. Thus, to compute transient response at different times, one has to compute each time only the $\underline{\Psi}$ vector and then the product of two vectors. Thus, this method requires approximately (n+N)M additions and multiplications, and M divisions, compared to n^2N required for the Liou method, where N is the number of points for which the transient response is evaluated.

Compared to Liou's method this [5] has the following advantages: (i) Fewer numerical operations; (ii) It does not require computation of the initial state vector; (iii) The time steps need not be spaced equally; (iv) Solution is not recursive as in the case of the Liou method; i.e., the response at time t_i is not a function of response at the previous time t_{i-1}.

We will first derive approximate expressions for the truncation error followed by a rigorous mathematical proof.

2. ERROR BOUND ON TRUNCATION ERROR

This paper is concerned with evaluating the following three major aspects of the transient response using new techniques and derivations: (i) The criterion for choosing the number of terms in $\exp(\bar{A}t)$ matrix series; (ii) The error bound if $\exp(\bar{A}t)$ matrix series is truncated after M terms; (iii) The basic limitations and range of applicability of the method. These major aspects have not been previously considered.

If we truncate $\exp(\bar{A}t)$ in (5) after M+1 terms

$$y_M(t) = [C_1, C_2, \ldots, C_n] \begin{bmatrix} 0 & 1 & \cdots & \beta_0 & \beta_1 \\ 0 & 0 & \cdots & \beta_1 & \beta_2 \\ 0 & 0 & \cdots & \beta_2 & . \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ . & . & \cdots & . & \beta_{n-1} \\ 1 & -a_1 & \cdots & \beta_{n-1} & \beta_n \end{bmatrix} \begin{bmatrix} 1 \\ t \\ t^2/2! \\ \vdots \\ t^{M-1}/(M-1)! \\ t^M/M! \end{bmatrix} \tag{8}$$

The terms contributing to (M+1)th the truncated term in the $\exp(\bar{A}t)$ series, are $\beta_1, \beta_2, \ldots, n$ and $t^M/M!$ in (8). Obviously the contribution to the transient response from the (M+1)th term is the following:

$$[C_1, C_2, \ldots, C_n] \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix} t^M/M!$$

i.e., $(C_1\beta_1 + C_2\beta_2 + \ldots + C_n\beta_n)\ t^M/M!$

Hence, when the magnitude of truncating the (M+1)th term is much smaller than the machine precision, then we can say that truncation error is given by:

$$|E_M^{(t)}| \simeq |C_1\beta_1 + C_2\beta_2 + \ldots + C_n\beta_n| \cdot \left|\frac{t^M}{M!}\right|$$

$$\simeq \left| \sum_{i=1}^{n} C_i\beta_i \right| \left|\frac{t^M}{M!}\right| \tag{9}$$

where β_i (i=1,...,n) are the elements of the (M+1)th column of the $\bar{\Phi}$ matrix. The truncation error at final simulation time (T) will be given by

$$|E_M^{(T)}| \simeq K \frac{T^M}{M!} \tag{10}$$

and

$$K = |\sum_{i=1}^{n} C_i \beta_i| \tag{11}$$

We conclude that (10) explains two points which are also observed in our numerical results: (i) The error bound is a function of time. The higher the value of T, the higher will be the number of terms required in the $\exp(\bar{A}t)$ series to achieve good accuracy. (ii) K is a function of elements of the $\underline{C}$ vector and the $\bar{A}$ matrix, i.e., the magnitudes of the coefficients of the numerator and denominator polynomials in (1). Thus, M increases as norm of $\bar{A}$ and $\underline{C}$ increases -- a result which corresponds to our numerical results in Section (3). In addition,

$$|E_M^{(T)}| << 1$$

when

$$M! >> KT^M$$

where K is a function of $f(\underline{C},\bar{A})$ and T is the finish time.

Let us rewrite (5) in truncated form as

$$y_M(t) = \underline{C} \sum_{j=0}^{M} \frac{(\bar{A}t)^j}{j!} \underline{B} \tag{12}$$

The truncation error will be given by

$$|y(t) - y_M(t)| = |\underline{C} \sum_{j=M+1}^{\infty} \frac{(\bar{A}t)^j}{j!} \underline{B}| \tag{13}$$

squaring both sides,

$$|y(t) - y_M(t)|^2 = |\underline{C} \sum_{j=M+1}^{\infty} \frac{(\bar{A}t)^j}{j!} \underline{B}|^2 \tag{14}$$

To get the bound on the truncation error, we use the Cauchy-Schwartz [9] inequality, and (14) reduces to:

$$|\varepsilon_M(t)|^2 \le ||\underline{C}||^2 \cdot || \sum_{j=M+1}^{\infty} \frac{(\bar{A}t)^j}{j!} \underline{B}||^2$$

$$|\varepsilon_M(T)|^2 \le ||\underline{C}||^2 \left(\sum_{j=M+i}^{\infty} \frac{(T||\bar{A}||)^j}{j!} \right)^2 \tag{15}$$

where T is the final time at which transient response is to be observed. The result in (15) is obtained using $||\underline{B}|| = 1$ and $||\bar{A}^j|| = ||\bar{A}||^j$. Consider the following equality

$$e^{||\bar{A}||T} = \sum_{j=0}^{M} \frac{(||\bar{A}||T)^j}{j} + \sum_{j=M+1}^{\infty} \frac{(||\bar{A}||T)^j}{j!} \tag{16}$$

We deduce

$$\sum_{j=M+1}^{\infty} \frac{(||\bar{A}||T)^j}{j!} = e^{||\bar{A}||T} - \sum_{j=0}^{M} \frac{(||\bar{A}||T)^j}{j!} \tag{17}$$

Hence from (15) and (17) we get

$$|\varepsilon_M| \le ||\underline{C}|| \sqrt{\left(e^{||\bar{A}||T} - \sum_{j=0}^{M} ||\bar{A}||^j \frac{(T)^j}{j!} \right)^2} \tag{18}$$

In the numerical evaluation of error bound given by (18), we must first note that the norm of n dimensional vector should have the following properties [9]: (i) $||\underline{X}|| \ge \phi$ for all Xs in linear space ν and $||\underline{X}|| = \phi$ iff $x = \phi$; (ii) $||\alpha\underline{X}|| = |\alpha| \cdot ||\underline{X}||$ for all scalars α and $\underline{X} \varepsilon \nu$; (iii) $||\underline{X}+\underline{Y}|| \le ||\underline{X}|| + ||\underline{Y}||$, which is the triangle inequality for all $\underline{X}$ and $\underline{Y}$ $\varepsilon\nu$ The infinity norm for any matrix $\bar{A}$ is defined by the absolute maximum row sum as follows:

$$||\bar{A}||_\infty \triangleq \max_i \sum_{j=1}^{n} a_{ij} \tag{19}$$

Similarly one norm for $\bar{A}$ is defined by the absolute maximum column sum

$$||\bar{A}||_1 \triangleq \max_j \sum_{i=1}^{n} |a_{ij}| \tag{20}$$

Let us now consider the evaluation of $|\varepsilon_M|$ in (18), the procedure which we propose is: (i) Using (20), evaluate $||\underline{C}||_1 \triangleq \sum_{i=1}^{n} |C_i|$; and (ii) $||\bar{A}||_1$, where $||\bar{A}||_1 = \max_j \{ \sum_{i=1}^{n} |a_{ij}| \}$, then evaluate $\exp(||\bar{A}||T)$; (iii) Evaluate the summation in (18) until $|\varepsilon_M|$ is less than the desired error. Record the value of M. The Cauchy-Schwartz inequality is used. We deduce from (18) that M increases as $||\bar{A}||$ increases; i.e., it is a function of the coefficients of the denominator polynomial. Also, M increases as T increases and $||\underline{C}||$ increases, i.e., M is a function of the final time and the coefficient of the numerator polynomial. Note that (10) and (18) seem different although the end results are the same.

3. COMPUTED RESULTS AND CONCLUSIONS

Consider the following fourth order problem as an example

$$y(s) = \frac{1}{(s+1)(s+1.5)(s+2)(s+2.5)}$$

Figure 1 shows the order of magnitude of the truncating term computed using (9) and actual observed error as a function of the number of terms in $\exp(\bar{A}t)$ series at time t = T = 10 sec.

From Figure 1 it is obvious that when the number of terms in the $\exp(\bar{A}t)$ series becomes large, then roundoff error exceeds the truncation error. Also, Simes and Mastascusa [7] have derived an expression for a bound on roundoff error in calculating the $e^{\bar{A}T}$ matrix (assuming M is such that truncation error is much smaller than machine precision -- M = 80 for the example under consideration). Graphical results are in good agreement with their error bound expression.

The graph also shows that the problem under discussion can be solved with good accuracy if the number of terms chosen in $\exp(\bar{A}t)$ matrix are greater than 80. But at that large value of M, roundoff errors dominate over the truncation errors because of the large numbers of computations and also because of propagation error in calculating the $\underline{\Psi}$ vector entries, viz. $t^k/k!$. This is observed in the upper curve of Figure 1.

Another observation from Figure 1 is that the error at first increases with M and then starts decreasing. This is because for t > 1 (t = 10 sec in this case) the entries in the $\underline{\Psi}$ vector build up in magnitude to a certain value of M (In our example $M = \hat{M}$, as marked on the figure.), and then starts decreasing in magnitude with further increase in M. In this case $\hat{M}$ is between 30 and 40. For a small truncation error, $M >> \hat{M}$ if t > 1.

If $||\bar{A}||$ is large and T is small or vice versa, the following normalization reduces the contribution due to roundoff errors:

$$y(t) = \underline{C}e^{\bar{A}t}\underline{B} = \underline{C}e^{(\bar{A}/x)tx}\underline{B} = \underline{C}e^{\bar{A}_N t_N}\underline{B}$$

where x is the normalization factor which is such that the norm of the $\bar{A}_N$ matrix is small. If t is already large and $||\bar{A}||$ is small, then x is a fraction. The purpose behind this normalization scheme, is to keep the relative magnitudes of the numbers in $\bar{\Phi}$ and $\underline{\Psi}$ matrix same.

From equality (18) we draw the family of curves (Figure 2) between $||\bar{A}||T$ versus M with $\varepsilon_N/||\underline{C}||$ as the dummy parameter. The fundamental use of these curves would be to estimate whether the given method can evaluate a system transient response to a desired accuracy. The numerical results compare favorably with the transform approach of Silverberg [6] and others [8].

4. ACKNOWLEDGEMENTS

The authors wish to thank Professor Norman I. Katz for his help during the initial stages of the research. They are also grateful to Professor Russell Pfeiffer for his continued encouragement and support.

5. REFERENCES

[1] M. L. Liou, Time and Frequency Domain Analysis of Linear Time-Invariant Systems, Chapter 4, Systems Analysis by Digital Computer, eds. F. F. Kuo and J. F. Kaiser, John Wiley and Co., 1966.

[2] M. L. Liou, "A Novel Method of Evaluating Transient Response," Proc. IEEE, Vol. 54, pp. 20-23, January 1966.

[3] W. E. Thompson, "Evaluation of Transient Response," Proc. IEEE, Vol. 54, p. 1584, 1966.

[4] J. Valand, "Calculation of Transient Response," Electr. Letters, Vol. 4, p. 261, June 1968.

[5] K. A. Muraleedharan and K. A. Stromsmoe, "On Computation of Transient Response," Proc. IEEE, Vol. 60, No. 8, p. 995, August 1972.

[6] M. Silverberg, "A New Method of Solving State Variable Equations Permitting Large Step Sizes," Proc. IEEE (letters), pp. 1352-1353, August 1968.

[7] J. G. Simes and E. J. Mastascusa, "Roundoff Errors in Calculating Exp (AT)", Proc. of NEC, Vol. 26, pp. 125-128, 1970.

[8] N. B. Rabbat, "Efficient Computation of the Transient Response of Lumped-Distributed Linear Networks," IEEE Proc. of the International Symposium on Circuits and Systems, San Francisco, April 1974.

[9] Issacson and Keller, "Analysis of Numerical Methods," New York, John Wiley and Co., 1966.

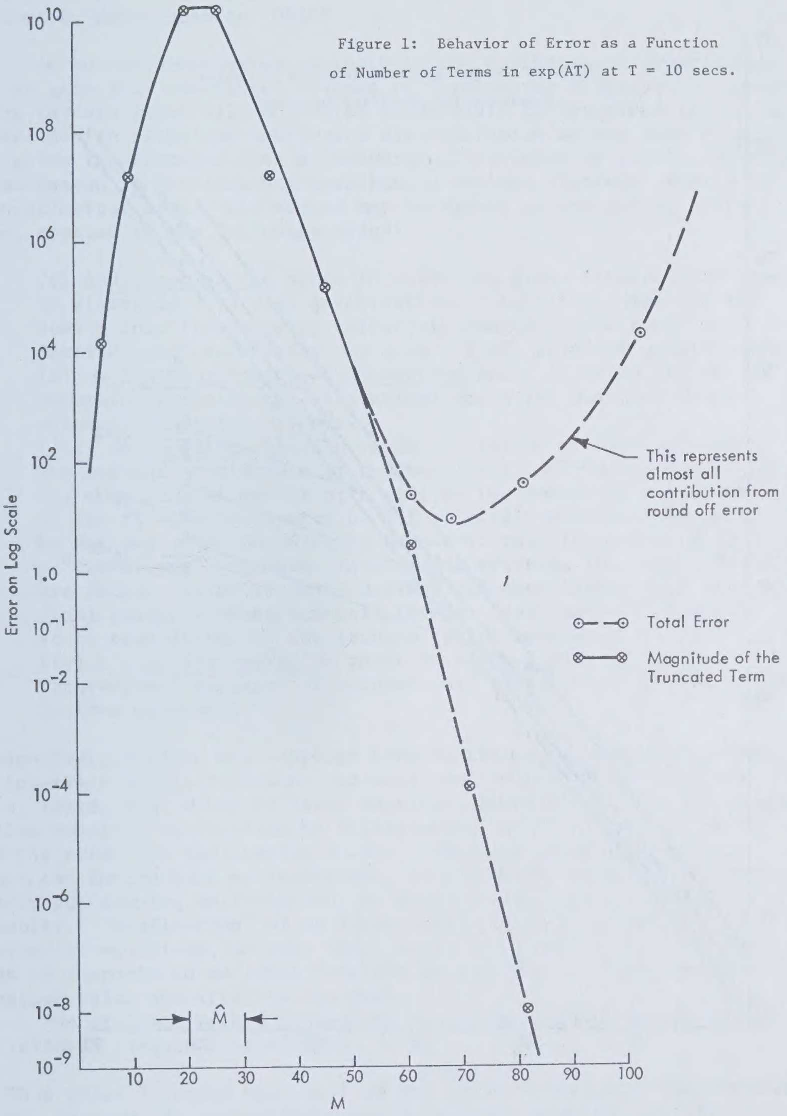


Figure 1: Behavior of Error as a Function of Number of Terms in $\exp(\bar{A}T)$ at T = 10 secs.

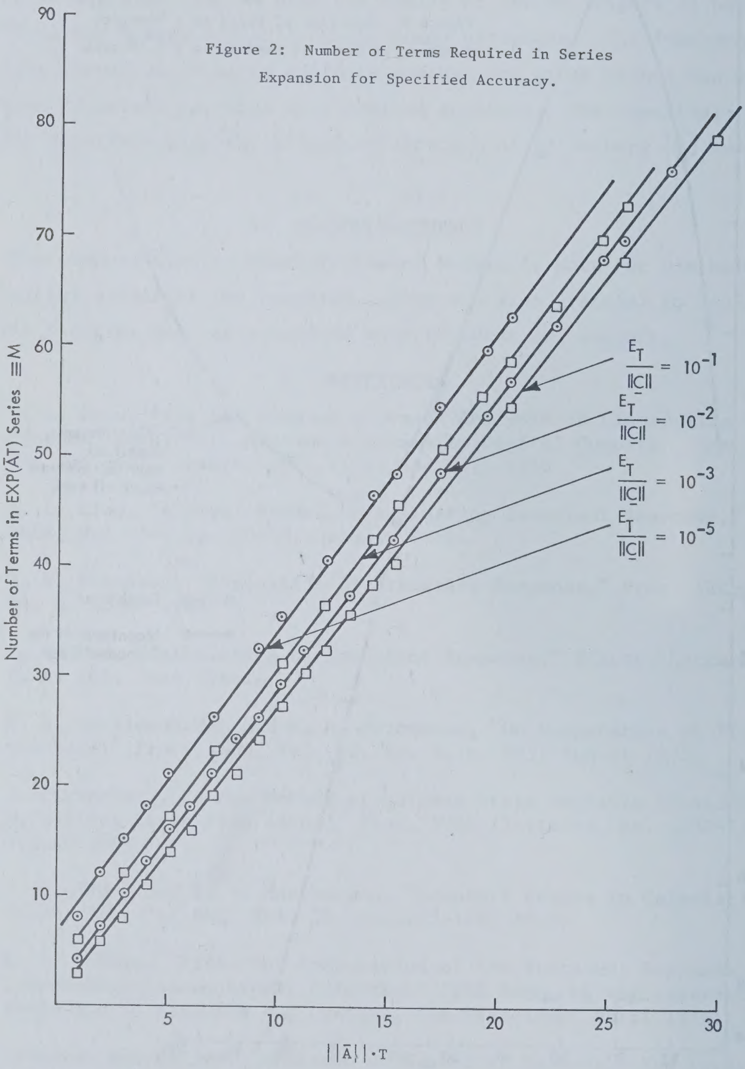


Figure 2: Number of Terms Required in Series Expansion for Specified Accuracy.

NON-LINEAR REALIZABILITY OF INPUT/OUTPUT TASKS: AN AUTOMATA-MOTIVATED SYNTHESIS APPROACH

MARTIN E. KALISKI
Department of Electrical Engineering
Northeastern University
Boston, Massachusetts 02115

An automata-motivated approach to the synthesis of input/output tasks by generally non-linear systems of first-order differential equations obeying certain physically-motivated constraints is described [1] . The proposed design technique involves a discretization scheme that constructs for a given input/output task a "convergent" sequence of finite automata, each automaton in the sequence realizing a derived discrete version of the input/output task. The method may be viewed as consisting of four phases, applied in the following order:

(i) A discretization phase in which the given input/output task is discretized through quantization of both its range and its domain into finite sets; infinitely many discrete tasks are derived, each successive one from a finer level of quantization.
(ii) A finite automaton construction phase in which finite automata realizing the discretized tasks are designed using automata-theoretic methods.
(iii) An encoding phase in which the states of each automaton are encoded as elements of n-dimensional real-vector space, for some natural number n; n will be the number of equations in the first-order system of differential equations that will be derived. The encoding is chosen so that the automata form a "convergent" sequence. Appropriate criteria for "convergence" are defined so as to force desired characteristics into the final design-- these criteria involve "derivatives" of the state transitions of the encoded finite automata.
(iv) A limiting operation phase in which limits of the "convergent" sequence of automata are taken so as to deduce the desired synthesis.

The input/output tasks are mappings with functions defined over finite real intervals as their domain and with real values as their range. The domain, range, and nature of these mappings are restricted in form so as to allow possible realization by differential equations, and so as to allow the potential application of the techniques proposed here. In essence, the inputs must be continuous, or piecewise constant, and the input/output mapping well-behaved, as implied by a general notion of continuity. "Realization" of an input/output task by a system of differential equations, as used here, means that the solution of the system in response to an input function in the domain of the task is the output value specified by the task.

The proposed method is applied to the semi-constructive design of a variable frequency oscillator.

[1] This paper is based upon part of the author's doctoral dissertation, "Dynamic Systems: An Automata-Motivated Analytic Approach",M.I.T. Electronics Systems Laboratory Report ESL-R-453, July, 1971. This work was supported in part by Office of Naval Research Contract-- N-00014-70-A-0362-0001.

BLOCK CODING FOR STATIONARY GAUSSIAN SOURCES WITH MEMORY UNDER A SQUARE-ERROR FIDELITY CRITERION [†]

HARRY H. TAN
Department of Electrical Engineering, Princeton University,
Princeton, New Jersey 08540

ABSTRACT

In this paper, we present a new version of the source coding theorem for the block coding of stationary Gaussian sources with memory under a square-error distortion criterion. The average square-error distortion of the optimum block source code of rate $R > R(D)$ is shown to decrease at least exponentially in blocklength to D, where $R(D)$ is the square-error criterion rate distortion function of the stationary Gaussian source with memory. The exponent of convergence of average distortion is explicitly derived.

I. INTRODUCTION

In 1959, the foundations of rate distortion theory was established by Shannon [1] when he defined the rate distortion function of an information source with respect to a fidelity criterion and proved the fundamental source coding theorems which give this function its operational significance. The standard source coding theorems give the existence of block codes of rate R and average distortion D when used to represent the information source, if and only if $R > R(D)$, the rate distortion function of the source. Thus in a communications system, where the codewords of the source code are properly channel coded for reliable transmission over the channel, $R(D)$ represents an absolute lower bound on the channel capacity required to achieve an overall system average distortion D. However, it is known that as the available channel capacity of the communication system approaches $R(D)$, the source code required to achieve this level of performance must have increasing blocklength. Since the complexity of the source encoder which implements the source code will necessarily increase with code blocklength, a reduction in channel capacity requirements entails a higher source encoder complexity. Thus, in the selection of source codes for a communication system, the tradeoff between channel capacity requirement and the source encoder complexity required to implement the source code should be considered. To resolve this problem, a study of the rate at which the average distortion of the optimum source code of rate $R > R(D)$ approaches D as its blocklength approaches infinity, is necessary.

Recently source coding theorems in the above context have been proved by Blahut [2], Omura [3], King [4] and Marton [5] for finite-alphabet discrete-time memoryless sources under bounded distortion measures. These works showed the exponential

---

[†] This research was supported by the National Science Foundation under Grant No. GK-42080.

rate of convergence of average distortion to D in blocklength n for optimum source codes of rate $R > R(D)$. In this paper a source coding theorem for the time discrete stationary Gaussian sources with memory under a square-error distortion criterion is proved which gives an upper bound to the rate of convergence of average distortion. This is a class of sources and distortion measure which play a fundamental role in many theoretical and practical problems.

Theorem 1 of Section II gives the main result of this paper, the source coding theorem for the discrete-time stationary Gaussian source under square-error distortion criterion. This theorem states that the average distortion of an optimum block source code of rate $R > R(D)$ decreases at least exponentially in blocklength to D. The proof of this theorem is given in Section III of this paper. The techniques we use in the proof are motivated by the methods developed by Omura [3].

II. THE SOURCE CODING THEOREM

We first consider a time-discrete stationary zero mean Gaussian source $\{X_t : t=0, \pm 1, ...\}$ where the random n=vector $\underline{X}_n = (X_1, X_2, ..., X_n)^T$ has covariance matrix $\underline{\Phi}_n = \{\varphi_{i-j}\}$ and where $\varphi_{i-j} = E[X_i X_j]$. We want to represent the source output with a sequence of reproduced letters from the real line according to an accuracy prescribed by a single-letter fidelity criteria generated by the mean square-error distortion criterion. That is, if a block of n source letters $\underline{x}^T = (x_1, ..., x_n)$ in $\mathcal{R}^n$ is represented by a block of reproduced letters $\underline{y}^T = (y_1, ..., y_n)$ in $\mathcal{R}^n$, a distortion $\rho_n(\underline{x}, \underline{y})$ given by

$$\rho_n(\underline{x}, \underline{y}) = \frac{1}{n} \sum_{i=1}^{n} (x_i - y_i)^2 \tag{1}$$

is incurred. For any integer $n \geq 1$ and any $R > 0$, a set $B^n = \{\underline{y}_1, ..., \underline{y}_k\}$ of reproduced vectors in $\mathcal{R}^n$ where $K = \lceil e^{nR} \rceil$ (i.e. K is an integer such that $K - 1 < e^{nR} \leq K$) is called a (n, R) source code. Such a code B^n is said to be of rate R and blocklength n. When the code B^n is used to represent the source output, each source word $\underline{x} \in \mathcal{R}^n$ is mapped into the codeword $\underline{y} \in B^n$ which minimizes $\rho_n(\underline{x}, \underline{y})$. The average distortion incurred when the code B^n is used to represent the source is defined to be

$$\rho_n(B^n) = E[\min_{\underline{y} \in B^n} \rho_n(\underline{X}_n, \underline{y})], \tag{2}$$

where the expectation is taken over the source ensemble. If $\rho_n(B^n) \leq D$, we say that the code B^n is D-admissible. In a communication system, the K codewords of a (n, R) source code B^n may be channel coded for reliable transmission over a channel of capacity C if and only if $R < C$. Thus the source coding problem is to determine for a prescribed average distortion level $D > 0$, the smallest rate of any D-admissible code. The solution to this problem is contained in Theorems 1 and 2 below which show that given a $D > 0$ and any $\varepsilon > 0$, a $(D+\varepsilon)$-admissible (n,R) code exists for sufficiently large n if $R > R(D)$ and that no D-admissible source code has rate less than $R(D)$, where $R(D)$ is the well-known mean-square error rate distortion function of the Gaussian source $\{X_t\}$ and is given parametrically [6] by

$$D_\theta = \frac{1}{2\pi} \int_{-\pi}^{\pi} \min(\theta, \Phi(\lambda))\, d\lambda \quad , \tag{3}$$

$$R(D_\theta) = \frac{1}{4\pi} \int_{-\pi}^{\pi} \max[0, \log \frac{\Phi(\lambda)}{\theta}]\, d\lambda \quad , \tag{4}$$

and where $\Phi(\lambda)$ is the spectral density function of the source $\{X_t\}$ and is given by

$$\Phi(\lambda) = \sum_{k=-\infty}^{\infty} \varphi_k e^{jk\lambda} \quad , \lambda \in [-\pi, \pi] . \tag{5}$$

Moreover the non-zero portion of the $R(D)$ curve is generated as the parameter θ in (3) and (4) traverses the interval $0 < \theta < \Delta = \text{ess sup } \Phi(\lambda)$ and $R(D) = 0$ for $D \geq D_{max} \overset{\Delta}{=} D_\theta \big|_{\theta=\Delta} = \varphi_0 = E[X_t^2]$.

For $D \geq D_{max}$, the positive side of the above assertion, that is the existence of a $(D+\varepsilon)$-admissible code of rate $R > R(D)$, is trivial since the code $B^1 = \{0\}$ is a D-admissible zero rate code. For $0 < D < D_{max}$, the following theorem holds. The proof of this theorem is given in Section III.

Theorem 1
Consider the discrete-time stationary zero mean Gaussian source $\{X_t, t=0, \pm 1, \ldots\}$ with spectral density function $\Phi(\lambda)$ given by (5) and mean square error criterion rate distortion function $R(D)$ given by (3) and (4). Then for each $D \in (0, D_{max})$, each $R \in (R(D), \infty)$ and each $\varepsilon_1 \in (0,D)$ and $\varepsilon_2 > 0$, there exist an integer $N(D,R,\varepsilon_1,\varepsilon_2)$ such that for every $n \geq N(D,R,\varepsilon_1,\varepsilon_2)$, there exist (n,R) source codes B^n with average distortion when used to represent the source bounded by

$$\rho_n(B^n) \leq D + \exp[-n(E(R, D-\varepsilon_1) - \varepsilon_2)] \ , \tag{6}$$

where for each fixed $D \in (0, D_{max})$, $E(R,D) > 0$ and is strictly convex in R for $R > R(D)$. Moreover for each $\theta \in (0, \Delta)$ and D_θ given by (3), $E(R, D_\theta)$ is given parametrically by

$$E(R_\rho, D_\theta) = -\rho \frac{\partial E_\infty(\rho,\theta)}{\partial \rho} + E_\infty(\rho,\theta) \ , \tag{7}$$

$$R_\rho = \frac{\partial E_\infty(\rho,\theta)}{\partial \rho} \quad , \tag{8}$$

where $\rho \in (-1,0)$ and $E_\infty(\rho,\theta)$ is given by

$$E_\infty(\rho,\theta) = -\frac{1}{4\pi} \int_{-\pi}^{\pi} \max\left[0, \rho \log \left(\frac{\theta(1+\rho)}{\Phi(\lambda)+\rho\theta}\right)\right] d\lambda \ . \tag{9}$$

The range of the parameter ρ in (7) and (8) corresponds to a range of rates $R_\rho \in (R(D_\theta), \infty)$ with $\frac{\partial R_\rho}{\partial \rho} < 0$.

The following converse source coding theorem follows from the variational definition of the rate distortion function ([7], p. 105) and can be established using an approach similar to that used by Berger [7, p. 281].

Theorem 2

There are no D-admissible codes of rate $R < R(D)$. ▮

Thus Theorem 1 has demonstrated the existence of (n,R) source codes B^n of rate $R > R(D)$ and average distortion $\rho_n(B^n)$ which decreases at least exponentially in blocklength n to D when used to represent a discrete-time Gaussian source under square-error distortion criterion. We note that this contrasts with the previous work of Bunin and Wolf [8] where the rate of convergence of average distortion of block codes of rate $R=R(D)$ was shown to be algebraic in blocklength for a time-discrete stationary first-order Gauss-Markov source under square-error distortion criterion. It is interesting to speculate whether this is the fastest possible rate of convergence in block coding of Gaussian sources under square-error distortion criterion. We would tend to conjecture that this is the fastest rate of convergence possible when $D > D_{crit}$, where $D_{crit} = D_\theta \big|_{\theta=\delta}$ and $\delta = \text{ess inf } \Phi(\lambda)$. For $D \leq D_{crit}$, there is less confidence in such a conjecture in view of the double exponential rate of convergence demonstrated by Omura and Shohara [9] in the block coding of certain symmetric finite-alphabet memory-less sources.

III. PROOF OF THEOREM 1

The proof of Theorem 1 is based on a random coding argument. For each D in $(0, D_{max})$ and R in $(R(D), \infty)$ we will exhibit a sequence of ensembles of (n,R) codes B^n for which the code ensemble average distortion satisfies the inequality (6) for sufficiently large n. The proof of the theorem then follows since there must be at least one sequence of codes B^n in this sequence of code ensembles with average distortion $\rho_n(B^n)$ satisfying (6) for sufficiently large n.

For $n \geq 1$, let $\{\lambda_k^{(n)} : k=1,2,\ldots n\}$ be the positive eigenvalues of $\underline{\Phi}_n$ and let $\underline{\Gamma}_n$ be the unitary matrix of orthonormal eigenvectors of $\underline{\Phi}_n$ so that $\underline{\Phi}_n = \underline{\Gamma}_n \underline{\Lambda}_n \underline{\Gamma}_n^T$ where $\underline{\Lambda}_n$ is the diagonal matrix $\{\lambda_i^{(n)} \delta_{ij}\}$. For $\theta \in (0, \Delta)$ and $R \in (R(D_\theta), \infty)$ consider an ensemble of (n,R) source codes $B^n = \{\underline{V}_1, \ldots, \underline{V}_K\}$ where $\underline{V}_k = \underline{\Gamma}_n \underline{Y}_k$ for $1 \leq k \leq K$ and where $\underline{Y}_K \equiv \underline{0}$ and $\{\underline{Y}_k\}_{k=1}^{K-1}$ are mutually independent random vectors independent of the source and having joint probability distribution $Q_n(\underline{y}) = \prod_{k=1}^{n} Q_{nk}(y_k)$ and where $Q_{nk}(y) = \int_{-\infty}^{y} \delta(t)\,dt$ if $\lambda_k^{(n)} \leq \theta$ and Q_{nk} is $N(\theta, \lambda_k^{(n)})$ if $\lambda_k^{(n)} > \theta$. Here $\delta(t)$ is the Dirac delta function. The expectation over the code ensemble of the average distortion incurred by codes in this ensemble when used to represent the source is called the code ensemble average distortion and is denoted by $\bar{\rho}_n$. We want to show that $\bar{\rho}_n$ satisfied the inequality (6) for sufficiently large n. Since $\underline{\Gamma}_n$ is a Euclidean distance preserving transformation, $\rho_n(\underline{X}_n,\underline{V}_k) = \rho_n(\underline{\tilde{X}}_n,\underline{Y}_k)$ where $\underline{\tilde{X}}_n = \underline{\Gamma}_n^T \underline{X}_n$ is a Gaussian random vector with joint probability density $\tilde{p}_n(\underline{x}) = \prod_{k=1}^{n} \tilde{p}_{nk}(x_k)$ where $\tilde{p}_{nk}(x)$ is a $N(0,\lambda_k^{(n)})$ probability density. Thus we have

$$\bar{\rho}_n = E[\min_{1\le k\le K} \rho_n(\underline{\tilde{X}}_n,\underline{Y}_k)] \quad , \tag{10}$$

where the expectation is over both source and code ensembles. Define the joint conditional probability distribution $Q_n'(\underline{y}|\underline{x}) = \prod_{k=1}^{n} Q_{nk}(y_k|x_k)$ where $Q_{nk}(y|x) = \int_{-\infty}^{y} \delta(t)dt$ if $\lambda_k^{(n)} \le \theta$ and $Q_{nk}(y|x)$ is $N(\beta_k^{(n)}x, \beta_k^{(n)}\theta)$ if $\lambda_k^{(n)} > \theta$, where $\beta_k^{(n)} = 1 - \theta/\lambda_k^{(n)}$. Define the joint conditional probability density $p_n(\underline{x}|\underline{y}) = \prod_{k=1}^{n} p_{nk}(x_k|y_k)$ where $p_{nk}(x|y)$ is $N(0,\lambda_k^{(n)})$ if $\lambda_k^{(n)} \le \theta$ and $N(y,\theta)$ if $\lambda_k^{(n)} > \theta$ for each y. It follows that

$$p_n(\underline{x}|\underline{y})dQ_n(\underline{y})\,d\underline{x} = \tilde{p}_n(\underline{x})dQ_n'(\underline{y}|\underline{x})d\underline{x}. \tag{11}$$

Define the functions $d_n : \mathcal{R}^{nK} \to \mathcal{R}$ and $g_n : \mathcal{R}^{n(K+1)} \to \mathcal{R}$ by

$$d_n(\underline{x},\underline{y}_1,\ldots,\underline{y}_{K-1}) = \min\{\rho_n(\underline{x},\underline{y}_1),\ldots,\rho_n(\underline{x},\underline{y}_{K-1}),\rho_n(\underline{x},\underline{0})\}, \tag{12}$$

and

$$g_n(x,y_0,\ldots,y_{K-1}) = \begin{cases} 1 & \text{if } \rho_n(\underline{x},\underline{y}_0) < d_n(\underline{x},\underline{y}_1,\ldots,\underline{y}_{K-1}) \\ 0 & \text{otherwise} \end{cases} \quad , \tag{13}$$

where $\underline{x},\underline{y}_0,\ldots,\underline{y}_{K-1} \in \mathcal{R}^n$. Since $(1-g_n)+g_n = 1$ and since $\int dQ_n'(\underline{y}_0|\underline{x}) = 1$ for every $\underline{x} \in \mathcal{R}^n$, it follows from (10) - (13) that

$$\begin{aligned} \bar{\rho}_n &= \int d_n(\underline{x},\underline{y}_1,\ldots,\underline{y}_{K-1})\Big[\prod_{k=1}^{K-1} dQ_n(\underline{y}_k)\Big]\tilde{p}_n(\underline{x})d\underline{x} \\ &= \int d_n[1-g_n]\Big[\prod_{k=1}^{K-1} dQ_n(\underline{y}_k)\Big]dQ_n'(\underline{y}_0|\underline{x})\tilde{p}_n(\underline{x})d\underline{x} \\ &\quad + \int d_n g_n\Big[\prod_{k=1}^{K-1} dQ_n(\underline{y}_k)\Big]dQ_n'(\underline{y}_0|\underline{x})\tilde{p}_n(\underline{x})d\underline{x} \\ &= \int d_n[1-g_n]p_n(\underline{x}|\underline{y}_0)\Big[\prod_{k=0}^{K-1} dQ_n(\underline{y}_k)\Big]d\underline{x} \\ &\quad + \int d_n g_n p_n(\underline{x}|\underline{y}_0)\Big[\prod_{k=0}^{K-1} dQ_n(\underline{y}_k)\Big]d\underline{x} \quad . \end{aligned} \tag{14}$$

Since $d_n[1-g_n] \le \rho_n(\underline{x},\underline{y}_0)$, the first integral in (14) can be bounded by

$$\int d\underline{x} \int \rho_n(\underline{x},\underline{y}_0)\,p_n(\underline{x}|\underline{y}_0)\,dQ_n(\underline{y}_0) = \frac{1}{n}\sum_{k=1}^{n} \min(\theta,\lambda_k^{(n)}). \tag{15}$$

Since $d_n \le \rho_n(\underline{x},\underline{0}) = \frac{1}{n}\sum_{k=1}^{n} x_k^2$ we have for a fixed $\underline{y}_1,\dots,\underline{y}_{K-1} \in \mathcal{R}^{n(K-1)}$ and an arbitrary $A > 0$,

$$\begin{aligned}
&\int d_n g_n p_n(\underline{x}|\underline{y}_0)\,dQ_n(\underline{y}_0)\,d\underline{x} \\
&\le \frac{1}{n}\sum_{k=1}^{n}\Big[\int_{\{x_k^2 \le A\}} x_k^2 g_n p_n(\underline{x}|\underline{y}_0)\,dQ_n(\underline{y}_0)\,d\underline{x} \\
&\qquad + \int_{\{x_k^2 > A\}} x_k^2 g_n p_n(\underline{x}|\underline{y}_0)\,dQ_n(\underline{y}_0)\,d\underline{x} \\
&\le A \int g_n p_n(\underline{x}|\underline{y}_0)\,dQ_n(\underline{y}_0)\,d\underline{x} \\
&\qquad + (\Delta + (2\Delta A/\pi)^{1/2}) \exp(-A/2\Delta) .
\end{aligned} \tag{16}$$

Now using an argument due to Omura [Lemma 1 of [3]], we can show that

$$\begin{aligned}
&\int g_n p_n(\underline{x}|\underline{y}_0)\Big[\prod_{k=0}^{K-1} dQ_n(\underline{y}_k)\Big]d\underline{x} \\
&\le \exp[-n\,(-\rho R + E_n(\rho,\theta))] ,
\end{aligned} \tag{17}$$

for any $\rho \in (-1,0)$ where

$$\begin{aligned}
E_n(\rho,\theta) &= -\frac{1}{n}\log\Big\{\int_{\mathcal{R}^n} d\underline{x}\,\Big[\int_{\mathcal{R}^n} p_n(\underline{x}|\underline{y})^{\frac{1}{1+\rho}} dQ_n(\underline{y})\Big]^{1+\rho}\Big\} \\
&= -\frac{1}{n}\sum_{k=1}^{n} \max\Big[0, \frac{\rho}{2}\log\Big(\frac{\theta(1+\rho)}{\lambda_k^{(n)}+\rho\theta}\Big)\Big].
\end{aligned} \tag{18}$$

Combining (14) - (18) with $A = n^2$, we have, for any $\rho \in (-1,0)$,

$$\bar{\rho}_n \le \frac{1}{n}\sum_{k=1}^{n} \min(\theta,\lambda_k^{(n)}) + \exp\Big[-n\Big(-\rho R + E_n(\rho,\theta) - O\Big(\frac{\log n}{n}\Big)\Big)\Big]. \tag{19}$$

Now, using the well known Toeplitz Distribution Theorem [10, p. 64], we conclude that $\frac{1}{n}\sum_{k=1}^{n} \min(\theta,\lambda_k^{(n)}) \to D_\theta$ and $E_n(\rho,\theta) \to E_\infty(\rho,\theta)$ as $n\to\infty$ where D_θ is given by (3) and $E_\infty(\rho,\theta)$ by (9). Thus using (19) we can conclude that for each $\theta \in (0,\Delta)$, $\rho \in (-1,0)$, $R \in (R(D_\theta),\infty)$ and each $\epsilon_1 > 0$, $\epsilon_2 > 0$ there exists an integer $N_1(\theta,\rho,R,\epsilon_1,\epsilon_2)$ such that for all $n \ge N_1$, we have

$$\bar{\rho}_n \le D_\theta + \epsilon_1 + \exp[-n(-\rho R + E_\infty(\rho,\theta) - \epsilon_2)]. \tag{20}$$

For fixed θ and R we can minimize the bound by maximizing the exponent $-\rho R+E_\infty(\rho,\theta)$ over the parameter $\rho \in (-1,0)$. Thus define for each $\theta \in (0,\Delta)$ and $R \in (R(D_\theta),\infty)$

$$E(R,D_\theta) = \sup_{\rho\in(-1,0)} [-\rho R + E_\infty(\rho,\theta)]. \tag{21}$$

The following lemma gives some useful properties of $E_\infty(\rho,\theta)$. The proof of this lemma is immediate and will be omitted here.

Lemma 1: For each $\theta \in (0,\Delta)$, $E_\infty(\rho,\theta)$ is strictly increasing and concave in ρ for $\rho\in(-1,0)$. Moreover $\lim\limits_{\rho\to 0} \dfrac{\partial E_\infty(\rho,\theta)}{\partial\rho} = R(D_\theta)$ and $\lim\limits_{\rho\to -1} \dfrac{\partial E_\infty(\rho,\theta)}{\partial\rho} = \infty$.

Since $E_\infty(\rho,\theta)$ is strictly concave in ρ for each θ, the maximization in (21) can be performed by differentiating $-\rho R + E_\infty(\rho,\theta)$ with respect to ρ and setting to zero whenever $R(D_\theta)<R<\infty$. Thus we have shown that $E(R,D_\theta)$ is given parametrically in ρ by (7) and (8). From Lemma 1, it is clear that the range of the parameter $\rho\in(-1,0)$ corresponds to a range of rates $R_\rho \in (R(D_\theta),\infty)$ in (7) and (8). The strict convexity of $E(R,D_\theta)$ in R for $R\in(R(D_\theta),\infty)$ is also immediate. We have thus shown that given any $\theta\in(0,\Delta)$, $R\in(R(D_\theta),\infty)$, $\epsilon_1 > 0$ and $\epsilon_2 > 0$, there exists an integer $N(D_\theta, R, \epsilon_1, \epsilon_2)$ such that for all $n \ge N(D_\theta, R, \epsilon_1, \epsilon_2)$.

$$\bar{\rho}_n \le D_\theta + \epsilon_1 + \exp[-n(E(R,D_\theta) - \epsilon_2)]. \tag{22}$$

Finally the inequality (22) may be written in the form of (6) by setting $D = D_\theta + \epsilon_1$ and appropriately restricting the range of ϵ_1. The proof of Theorem 1 is now complete since there must exist at least one sequence of codes B^n in the sequence of code ensembles whose average distortion $\rho_n(B^n)$ satisfies the inequality (22) when $n \ge N(D_\theta, R, \epsilon_1, \epsilon_2)$.

IV. CONCLUSION

This paper has presented a new version of the source coding theorem for the block coding of stationary time discrete Gaussian sources with memory under a square-error distortion criterion. The average square-error distortion of the optimum block source code of rate $R > R(D)$ was shown to decrease at least exponentially in blocklength to D. In both cases the exponent of convergence of average distortion was explicitly derived. These results have application in the study of tradeoffs between source encoder complexity and channel capacity requirements in a communication system designed to transmit data from a stationary Gaussian source.

We note that Theorem 1 can be immediately generalized to continuous-time Gaussian sources under integral square-error distortion criterion by using a Karhunen-Loeve expansion of the source. The square-error distortion criterion can also be generalized to frequency or time-weighted square-error distortion criteria as defined in [11]. Theorem 1 can also be generalized to homogeneous Gaussian random fields under either integral square-error or weighted integral square-error distortion criterion as defined in [12]. Finally we note that an interesting open problem is to determine lower bounds on the rate of convergence of average square-error distortion of the optimum block source code for Gaussian sources.

REFERENCES

1. Shannon, C.E. (1959), Coding Theorems for a Discrete Source with a Fidelity Criterion, IRE National Convention Record, New York, Part 4, 142-163.
2. Blahut, R.E. (1972), An Hypothesis Testing Approach to Information Theory, Ph.D. Dissertation, Electrical Engineering Department, Cornell University, Ithaca, New York.
3. Omura, J.K. (1973), A Coding Theorem for Discrete-Time Sources, IEEE Trans. on Inf. Theory, PGIT-19, 490-498.
4. King, W.C. (1973), Block Coding Theorems for Discrete Memoryless Channels and Discrete Memoryless Sources, Ph.D. Dissertation, System Science Department, School of Engineering and Applied Science, University of California, Los Angeles, California.
5. Marton, K. (1974), Error Exponent for Source Coding with a Fidelity Criterion, IEEE Trans. on Inf. Theory, PGIT-20, 197-199.
6. Kolmogorov, A.N. (1956), On the Shannon Theory of Information in the Case of Continuous Signals, IRE Trans. on Inf. Theory, PGIT-2, 102-108.
7. Berger, T. (1971), Rate Distortion Theory - A Mathematical Basis for Data Compression, Prentice Hall, Englewood Cliffs, New Jersey.
8. Bunin, B.J. and J.K. Wolf (1971), Convergence to the Rate-Distortion Function for Gaussian Sources, IEEE Trans. on Inf. Theory, PGIT-17, 65-70.
9. Omura, J.K. and A. Shohara (1973), On Convergence of Distortion for Block and Tree Encoding of Symmetric Sources, IEEE Trans. on Inf. Theory, PGIT-19, 573-577.
10. Grenander, U. and G. Szego (1958), Toeplitz Forms and Their Applications, University of California Press, Berkeley, California.
11. Sakrison, D.J. (1968), The Rate Distortion Function of a Gaussian Process with a Weighted Square Error Criterion, IEEE Trans. on Inf. Theory, PGIT-14, 506-508.

12. Sakrison, D.J. and V.R. Algazi (1971), Comparison of Line-by-Line and Two-Dimensional Encoding of Random Images, IEEE Trans. on Inf. Theory, PGIT-17, 386-398.

OPTIMALITY OF DELTA MODULATOR

G. THOMAS AND V.V.S. SARMA
Department of Electrical Communication Engineering
Indian Institute of Science
Bangalore-560012, INDIA.

ABSTRACT

In this paper we point out certain structural similarities between optimal tracking control systems and optimal differential encoders, and show via an application of Pontryagin's minimum principle that simple delta modulation schemes have the property of minimum time signal acquisition. Cases of delta modulators with leaky and double-integration are also studied.

INTRODUCTION

Recently there has been considerable interest in theoretical studies on delta modulation (DM)[1] which has proved itself to be a simple practical digital encoding scheme. Several developments in DM are summarised in a recent review paper[2]. It has also been recognized that DM is a non-linear sampled data control system[3]. Wang[4] formulated the sufficient conditions for stability of DM using the Popov-Tsypkin theorems. Idling patterns and limit cycling problems were studied by Wang[5] and Smith[6]. A more recent and interesting approach is due to Wang[7] wherein Pontryagin's minimum principle[8] is used to design an optimal digital encoder satisfying a minimum mean squared error (MMSE) criterion over one sampling interval. Wang's scheme results in a rather complex system in relation to the basic DM system in Fig.1. In this note we show that there are systems comparable in simplicity to the basic DM scheme, which have some optimality properties as well. The DM is viewed as a tracking control loop. With reference to Fig. 1, the forward path element is a controller which makes the output x(t) of the feedback path element track the signal s(t). A suitable performance measure is chosen and the corresponding optimal controller is derived using Pontryagin's minimum principle.

PROBLEM FORMULATION

The basic DM scheme is shown[9] in Fig. 1. The hold circuit keeps the input constant over a sampling interval and the system tries to acquire it in this period. It may however be noted that essentially equivalent configurations with the sampler located in the loop have been used[2].

The first step in the present approach to digital encoding is to establish explicitly a congruence between the optimal tracking control problem and the optimal differential encoding problem for a deterministic signal using a known feedback plant. Consider the basic deterministic optimal tracking control scheme shown in Fig. 2. The plant $H(s)$ is to track the signal $s(t)$ in such a way as to satisfy some specified performance measure. The problem is to specify the optimal controller N that will operate on the instantaneous tracking error, $e(t) = s(t) - x(t)$ to synthesise the optimal instantaneous control signal $u(t)$ subject to any amplitude constraints on $u(t)$.

If we view the same scheme in a different perspective, we can restate the problem as follows: Given a deterministic signal $s(t)$, and a specified plant $H(s)$, what is the optimal channel signal $u(t)$, (and therefore, what is N?) which when applied to the plant $H(s)$, at the (remote) decoding station, will make $x(t)$, the output of $H(s)$, track $s(t)$ in such a way as to minimise a chosen cost measure. The mathematical equivalence of the two problems is now obvious. The different variations of the problem have been fairly exhaustively treated in the control-theory context[8].

Amplitude constraints on $u(t)$ of the form, $|u(t)| \leq$, can be brought into the communications problem for the same reasons as in the control context, viz., it is practically unrealistic to attempt to realise arbitrarily large signals. All electronic systems, for example, saturate beyond a certain signal range. But we wish to note here, in anticipation of the results in the next section, that such a constraint is particularly rewarding in the communications context since the resulting optimal channel signal is inherently in the binary form.

We treat the tracking problem as a succession of interval-wise acquisition problems. The signal s(t) is sampled at a sufficiently high rate so that sample-to-sample variation is small, and this sampled value is held constant over the ensuing sampling interval. In this period x(t) has to catch up with $s_1(t)$. Also, it is already noted that the MMSE criterion gives rise to a complicated digital encoder. It is worthwhile to study the well-known minimum time criterion in optimal control theory, in the communications context.

SOLUTIONS FOR OPTIMAL CHANNEL SIGNALS

In this section, we consider three variations of the DM configuration used in practice. In each case the interval-wise minimum time control problem is studied via minimum principle.

(i) The Single Integrator Plant:-

This corresponds to the case when H(s) in Fig. 1 is a single integrator. The state equation is

$$\dot{x} = u \qquad \ldots \qquad (1)$$

It is required to minimise

$$\int_0^{T_a} dt, \text{ subject to } |u(t)| \leq \Delta$$

where T_a is the time of acquisition.

For the time-optimal problem, the Hamiltonian H is written in terms of the constate p as

$$H = 1 + pu \qquad \ldots \qquad (2)$$

The costate equation is

$$\dot{p} = \partial H/\partial x = 0 ,$$

with the solution

$$p(t) = \text{constant} = p(0) \qquad \dots \quad (3)$$

The minimum time solution is[8]

$$\overset{*}{u}(t) = - \Delta \text{ sgn } p(t) = - \Delta \text{ sgn } p(0) \qquad \dots \quad (4)$$

The minimum time solution is either $+\Delta$ or $-\Delta$ throughout and there are no switchings in polarity.

Now refer to Fig. 3 which shows the state-time trajectory of the integrator output over one interval. $x(nT)$ is the initial state of the system at $t = nT$. The target signal $s(t)$ is sampled and held constant over the ensuing interval. Since the input is $\pm \Delta$, the output is a positive or negative going ramp. Therefore the choice between $\pm \Delta$ is uniquely decided by the law,

$$\begin{aligned} \overset{*}{u}(t) &= + \Delta \text{ if } \left[s(nT) - x(t)\right] \geq 0 \\ &= - \Delta \text{ if } \left[s(nT) - x(t)\right] < 0 \end{aligned}$$

which is the familiar DM decision scheme:

$$\overset{*}{u}(t) = \Delta \text{ sgn } \left[s(nT) - x(t)\right] \quad nT \leq t \leq (n+1)T \qquad \dots \quad (5)$$

Thus the optimal encoder (Fig.2) is really the DM as shown in Fig. 1 in the sense of minimum-time signal acquisition.

It is seen from equation (1) that the minimum time T_{min} required to make the state-transition from $x(t) = x(nT)$ to $x(t) = s(nT)$ is given by

$$T_{min} = |s(nT) - x(nT)|/\Delta \qquad \dots \quad (6)$$

(ii) First-Order Plant with Time-Constant:

We now replace the pure integrator $1/s$ by a plant of the form$(1/s+a)$ where a is real and positive.

The system equation in this case is

$$\dot{x} = -ax + u \qquad \ldots (7)$$

The Hamiltonian is obtained as

$$H = 1 + p\,(-ax + u) \qquad \ldots (8)$$

and the costate differential equation is

$$-\partial H/\partial x = \dot{p} = -ap \qquad \ldots (9)$$

Therefore, if $p(t) = p(0)$ at $t = 0$,

$$p(t) = p(0)\,\exp\,(-at) \qquad \ldots (10)$$

The optimal solution is

$$\begin{aligned} \overset{*}{u}\,(t) &= -\,\Delta\,\mathrm{sgn}\;p(t) \\ &= \left[-\,\Delta\,\mathrm{sgn}\;p(0)\right]\left[\mathrm{sgn}\quad \exp\quad(-at)\right] \\ &= -\,\Delta\,\mathrm{sgn}\;p(0) \end{aligned} \qquad \ldots (11)$$

Therefore, $\dot{x} = -ax \pm \Delta = -\,ax + D$ (say) $\qquad \ldots (12)$

It can be shown directly from (12) that for an initial state $x(nT)$, the subsequent state trajectory under the control as in (11) is given by

$$x(t) = \frac{D}{a}\left[\ 1-(1 - \frac{a}{D}\,x(nT))\,\exp\,(-at)\right] \qquad \ldots (13)$$

Thus the minimum-time state trajectory is an exponential curve starting from $x(t) = x(nT)$ at $t = nT$ and reaching $x(t) = \frac{D}{a} = \pm\,\Delta/a$ in infinite time. Fig. 4 shows the state-time trajectories. It is clear that so long as $x(nT)$ and $s(nT)$ are within the range $\left[-\,\Delta/a,\ +\,\Delta/a\right]$ the control exists, is unique, and is given by

$$\overset{*}{u}(t) = \mathrm{sgn}\left[\ s(nT) - x(t)\right] \qquad \ldots (14)$$

In fact, if the target $s(nT)$ is within $\pm$ (Δ/a) range, then (14) holds irrespective of the initial value $x(nT)$. Since the conditional probability density of the next-sample amplitude is usually symmetric about the 'present' sample[10], a good design strategy is to choose (Δ/a) large enough to accommodate the largest input magnitude and then to ensure this bound by limiting at the input. Thus again we see that for signal variations within the allowable range $\left[-\Delta/a\,,\ \Delta/a\right]$, for the 1/(s + a) plant also, the minimum-time strategy is a DM scheme as suggested by (14).

To show that the 1/(s+a) plant is faster in signal acquisition than the 1/s plant, we compute the minimum time T_{min} required to effect the state transition from x(nT) to s(nT). From equation (13)

$$\frac{D}{a}\left[1-\left(1-\frac{a}{D}\,x\,(nT)\right)\exp\left(-a\,T_{min}\right)\right] = s(nT)$$

and

$$T_{min} = \frac{1}{a}\ln\left\{\left[1-\frac{a}{D}\,x(nT)\right]\Big/\left[1-\frac{a}{D}\;s(nT)\right]\right\} \qquad \ldots \quad (15)$$

For example, in the simple case where s(nT) = 0, we have

$$T_{min} = \frac{1}{a}\ln\left[1+\ a\ |\,x\ (nT)\,|\ /\Delta\right] \qquad \ldots \quad (16)$$

using the inequality

$$\ln\,(1+x) \leq x \qquad \ldots \quad (17)$$

We have

$$\text{RHS of } (16) \leq |x(nT)|/\Delta = (T_{min} \text{ for single integrator}) \qquad \ldots \quad (18)$$

(iii) DM with Double Integration:

H(s) in Fig. 1 is now assumed to be $1/s^2$. For this case[8], the minimum time trajectories are generated by $\overset{*}{u}(t) = \pm\Delta$ and they form a pair of directed parabolas as in Fig. 5. The target set is the line x_1 = s(nT). Now with reference to Fig. 5, suppose that s(nT) > x(nT). If s(nT) lies outside the shaded region (generated by the x_1 - axis intercepts of the parabolas passing through x(nT)), it is clear that there is only one unique path, under + Δ, which leads from X(nT), to x_1 = s(nT). If x_1 = s(nT) happens to be within the shaded region (and yet s(nT) > x(nT)) then both the parabolas starting from X(nT) will eventually meet the line x_1 = s(nT). But it is easy to see that the + Δ path is the minimum-time one. Likewise for s(nT) the optimal control is $\overset{*}{u} = -\Delta$. These observations hold good for any location of the initial and target-values. Thus in general, the time-optimal strategy for moving from any X(nT) to any $x_1 = s(nT)$ is given as

$$\overset{*}{u}(t) = \Delta \operatorname{sgn}\left[s(nT) - x(t)\right] \qquad \ldots (19)$$

which, of course, is the DM scheme.

CONCLUSIONS

Traditional designs of signal encoding schemes concentrate mainly on signal properties but the characteristics such as stability and optimality of the encoding system are often neglected. This gave rise to problems in the hardware implementation and adhoc schemes[10] were devised to correct these system problems. This paper presents a new view point of looking at the system which does the signal encoding task, utilizing the modern developments in control and system theory. The simple example considered in the paper shows that the ordinary delta modulators has minimum time signal acquisition property. Also the problems of slope overload and granular noise are clearly seen on the phase plane. It is also instructive to use other performance criteria such as 'fuel-optimality' which is well known in optimal control literature. The performance index to be minimized is $\int_{nT}^{(n+1)T} |u(t)|dt$. In the communications context this is the minimum absolute noise criterion, considered by Gersho[11] in DM studies. This gives rise to a relay with dead zone and may lead to data compression.

ACKNOWLEDGEMENT

The authors are indebted to Prof. B.S. Ramakrishna, Professor-in-charge of the Centre for Information Processing, for his keen interest and helpful suggestions during the course of the work reported here. They wish to thank their colleagues Drs. N. Viswanatham, B. Yegnanarayana and D.N. Dutt for helpful comments. They gratefully acknowledge the

support of the Centre for Information Processing for this study.

REFERENCES

1. F. dejagar, 'Delta modulation, a method of PCM transmission using a one-unit code' Philips Research Report, Vol.7, 1953, pp. 442-466.

2. H.R. Schindler, 'Delta Modulation', IEEE Spectrum, 1970, pp. 69-78.

3. G.J. Marphy and G. Shyraym, 'Delta modulation in feedback control', Proc. Natl. Elec.Conf., 1965, pp. 609-612.

4. P.P. Wang, 'An absolute stability criterion for delta modulation' IEEE Trans. Comm. Syst., 1968, pp.186-188.

5. P.P. Wang 'Idle channel noise in delta modulation', IEEE Trans. Comm. Syst., 1968, pp. 737-742.

6. B.M. Smith, 'A note on the idling pattern of a delta modulated control system', Australian Telecom. Research, ATR Vol. 5, No.2, 1971, pp. 23-26.

7. P.P. Wang, 'The optimality of variable sampling schemes for a digital encoder' Proc. JACC, Washington Univ., 1971.

8. M. Athans and P.L. Falb, 'Optimal Control', McGraw-Hill Book Co., 1966.

9. J.B. O'Neal, 'Predictive quantising systems for transmission of TV signals', BSTJ, Vol. 45, 1966, pp.689-721.

10. N.S. Jayant, 'Adaptive delta modulation with one-bit memory', BSTJ, Vol. 49, 1970, pp. 321-342.

11. Allen Gersho, 'Stochastic stability of delta modulation' BSTJ, Vol. 51, No.4, 1972, pp. 821-842.

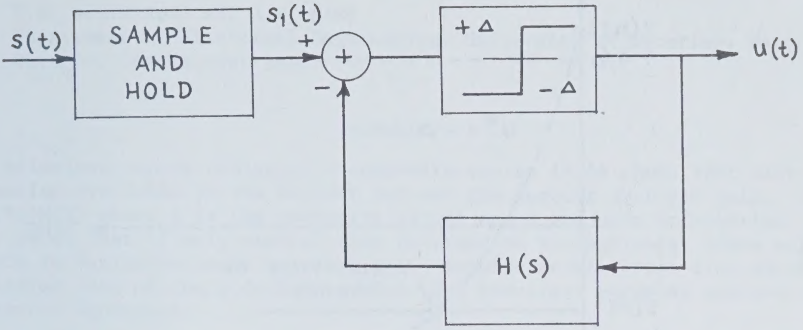

s(t)
SAMPLE AND HOLD
$s_1(t)$
+
−
+
+Δ
−Δ
u(t)
H(s)

FIG. 1 BASIC DM SCHEME.

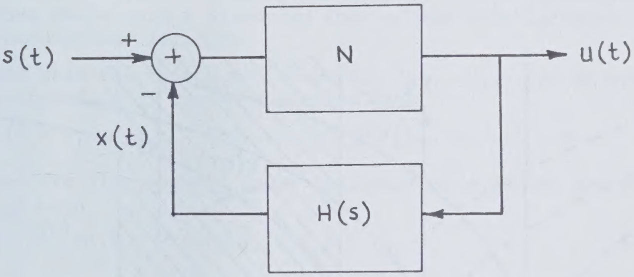


FIG. 2 TRACKING CONTROL SYSTEM.

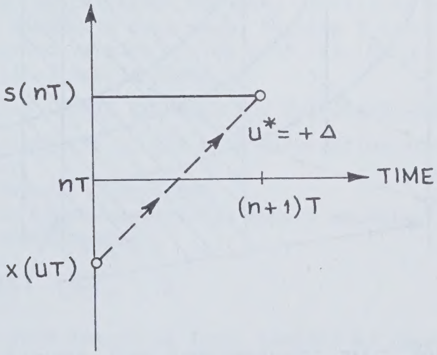

s(nT)
$u^* = +\Delta$
nT
TIME
(n+1)T
x(uT)

FIG. 3 STATE-TIME TRAJECTORY FOR SINGLE INTEGRATION.

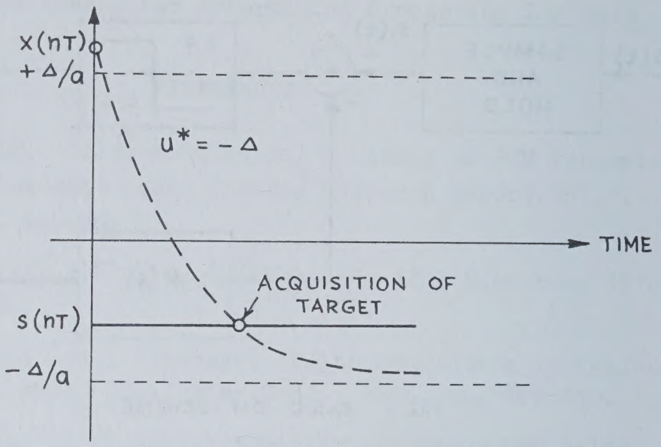

x(nT)
+Δ/a
u* = −Δ
TIME
ACQUISITION OF TARGET
s(nT)
−Δ/a

FIG. 4 STATE-TIME TRAJECTORY FOR LEAKEY INTEGRATION.

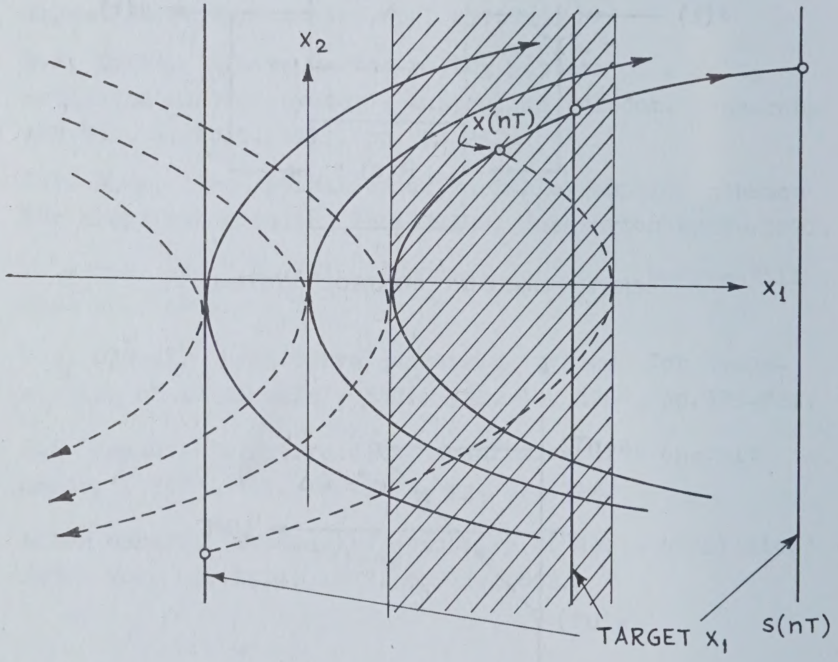

x₂
x(nT)
x₁
TARGET x₁
s(nT)

FIG. 5. MINIMUM-TIME TRAJECTORIES FOR DOUBLE INTEGRATION.

VARIABLE-LENGTH ENCODING FOR DISCRETE MEMORYLESS COMPOSITE SOURCES

P.S. BUDIHARDJO AND J.W. MARK
Department of Electrical Engineering, University of Waterloo, Waterloo, Ontario, Canada, N2L 3G1

ABSTRACT

For noiseless source coding of a composite source it is shown that side information available to the encoder but not the decoder does not help, i.e., $H(X,S) \geq H(X)$ where X is the composite source and S the side information. It is also shown that if only partial side information is available, there might be a gain in variable-length encoding the composite source while also sending a distorted form of the side information. The resultant encoding process would be source dependent.

1. INTRODUCTION

In this paper we will consider block-to-variable-length encoding for discrete memoryless composite sources. A composite source is a collection of subsources, where at each time instant one of the subsource is selected according to some switch probability[1](Fig. 1). The exact information on the switch positions is called the side information. We are particularly interested in the case where only a distorted form of the side information is available at the encoder and decoder.

We will assume that the switch positions b_i, i=1,..., h are independent identically distributed (iid) random variables with

$$\Pr\{S = b_i\} = q_i, \quad i=1,\ldots, h \; ; \; b_i \in \{b_1,\ldots,b_h\} = B \qquad \ldots. \; (1)$$

The subsources are also assumed to be iid, sharing a common source alphabet $A=\{a_1,\ldots, a_M\}$ with

$$\Pr \{Z^{(i)}=a_j\} = \Pr \{X=a_j/S=b_i\} = P_{ij} \qquad \ldots. \; (2)$$
$$i = 1,\ldots, h$$
$$j = 1,\ldots, M$$

The source encoding scheme is block-to-variable-length, i.e., to each source sequence of length N, $\underline{X}=(X_1,\ldots,X_N)$ we assign a code word of length $n(\underline{X})$.

Section 2 discusses three different encoding situations and lower bounds for the average codeword length in each case. Section 3 describes a coding scheme in which only a distorted version of the side information is available. This scheme may have practical applications.

2. VARIABLE-LENGTH ENCODING FOR COMPOSITE SOURCES

Depending on the availability of the side information at the encoder or decoder we will consider

Case I: Encoder and decoder uninformed

The problem reduces to block-to-variable-length encoding for a simple iid source with mixture distribution

$$\tilde{P}_j = \sum_{i=1}^{h} q_i P_{ij} \quad ,j=1,\ldots,M \qquad \ldots. \; (3)$$

and the average code word length is lower bounded by (see Appendix I).

$$\bar{\tilde{n}} \geq \frac{H(X)}{\log Q_c}$$

where H(X) is the entropy of the sequence X.

$$> \frac{\sum_{i=1}^{h} H(X/S=b_i)\, q_i}{\log O_c} \qquad \text{.... (4)}$$

$$\text{or} \quad \tilde{n} > \frac{H(X/S)}{\log Q_c} \qquad \text{.... (5)}$$

where Q_c is the size of the code alphabet.
Case II: Encoder and decoder informed
If the side information is available to both encoder and decoder, clearly the average code word length is bounded by[2].

$$\frac{H(X/S)}{\log Q_c} \le \bar{n}_s < \frac{H(X/S)}{\log Q_c} + 1 \qquad \text{.... (6)}$$

In Appendix II we have shown that this is equivalent to a coding scheme, where to each source symbol a_i we assign a code word of length

$$n_j = \sum_{i=1}^{h} n_{i,j} \quad \Pr\{S=b_i/X=a_j\} \qquad \text{.... (7)}$$

$$j = 1,\ldots,M$$

where $n_{i,j}$ is the optimal code word length for the symbol a_j of subsource i, i=1,...,h and

$$\Pr\{S=b_i/X=a_j\} = \frac{q_i\, P_{ij}}{\sum_{i=1}^{h} q_i\, P_{ij}} \qquad \text{.... (8)}$$

Case III: Encoder informed, decoder uninformed
Since the side information is not available at the decoder, we have to supply this information to the decoder if the scheme in case II is to be used. If variable-length encoding is to be used for the side information the average code word length will be increased by

$$\Delta \ge \frac{H(S)}{\log Q_c} \qquad \text{.... (9)}$$

resulting in an overall average code word length of

$$\hat{\bar{n}} = \bar{n}_s + \Delta$$

$$\ge \frac{H(X/S)}{\log Q_c} + \frac{H(S)}{\log Q_c}$$

$$\text{or} \quad \hat{\bar{n}} \ge \frac{H(X,S)}{\log Q_c} \qquad \text{.... (10)}$$

However,

$$H(X,S) = H(X) + H(S/X)$$

$$\text{or} \quad H(X,S) \ge H(X) \qquad \text{.... (11)}$$

thus,

$$\hat{\bar{n}} \ge \frac{H(X)}{\log Q_c} \qquad \text{.... (12)}$$

which means that we might as well ignore the side information that is available to us at the encoder.

3. ENCODING WITH PARTIAL SIDE INFORMATION

In section 2 we have shown that if we have to send the side information that is available at the encoder to the decoder, then the average code word length will be increased to the point where we might as well ignore the side information. In this section, however, we will consider the case where only a distorted form of the side information is available at the encoder and the decoder is uninformed.

Consider the block diagram in Fig. 2. From the data processing theorem[1], we have

$$I(X;\hat{S}) \geq I(S;\hat{S}) \geq R_s(D) \qquad \ldots. \ (13)$$

where $R_s(D)$ is the rate distortion function for the side information S_k. Note that the first inequality in (13) reduces to equality only if the first block in Fig. 2 is lossless, i.e.

$$P(S/X, \hat{S}) = P(S/\hat{S}) \qquad \ldots. \ (14)$$

and $P(S,\hat{S}) \neq 0$ .... (15)

Now,

$$H(X/\hat{S}) = -I(X;\hat{S}) + H(X)$$

or $H(X/\hat{S}) \leq H(X) - R_s(D)$ .... (16)

and

$$\frac{H(X/\hat{S})}{\log Q_c} + \frac{R_s(D\ )}{\log Q_c} \leq \frac{H(X)}{\log Q_c} \qquad \ldots. \ (17)$$

Note that the LHS of (17) is just the lower bound of the average code word length required if a distorted form of the side information is available at the encoder plus the average code word length required to send the distorted side information to the decoder

In most cases, however, the side information is not available to us, i.e., we have to estimate the side information sequence based on the observation of the source sequence. In an ideal case where the side information is available at the encoder, the side information coding process, C_N, is just a many to one mapping:

$$f_s \ : \ B^N \longrightarrow M = \{\underline{m}_1, \ldots, \underline{m}_K\} \qquad \ldots. \ (18)$$

where $m_i \ \varepsilon \ B^N$. The side information sequence $\underline{S}$ is encoded into the code word m_i such that

$$d(\underline{m}_i, \underline{S}) = \min_{j=1,\ldots,K} d(\underline{m}_j, \underline{S}) \qquad \ldots. \ (19)$$

Thus, the coding process can be regarded as a rule for dividing the subspace B^N into K disjoint subspaces $\{W_i,\ i=1,\ldots,K\}$ where each subspace corresponds to a particular code word m_i.

For each code word m_i, $i=1,\ldots,K$ we have a variable-length encoder $(\mathrm{VLE})_i$ which is optimum in the sense that

$$H(\underline{X}/\underline{m}_i) \leq \bar{n}_N^{(i)} = \sum_x n^{(i)}(\underline{X})p(\underline{X}/m_i) < H(\underline{X}/\underline{m}_i) + 1 \qquad \ldots. \ (20)$$

If the VLE receives the sequence $\underline{X}$ and the side information happens to be $\underline{S} \ \varepsilon \ W_i$, then

$$n_N^{(i)}(\underline{X}) \leq n_N^{(j)}(\underline{X}) \qquad , \ i \neq j, \quad i,j = 1,\ldots,K \qquad \ldots. \ (21)$$

Thus, in practice we can "estimate" the side information sequence by observing the shortest code word length produced by the VLE's. (See Fig. 3).

Let $\underline{V}_i$ be a sequence of length N such that

$$d(\underline{V}_i, \underline{m}_i) = 0 \qquad \forall\ i \qquad \ldots. \ (22)$$

Then for a given $\underline{S} \ \varepsilon \ W_i$, but $\underline{S} \neq V_i$ the code word length will increase. The average code word length when $\underline{S} \ \varepsilon \ W_i$ is

$$\bar{n}_N^{(i)} = \sum_{\underline{X}} \ \sum_{\underline{S} \ \varepsilon \ W_i} p(\underline{X}, \underline{S}(m_i) \ n_N^{(i)}(\underline{X}) \qquad \ldots. \ (23)$$

In the optimum case

$$n_N^{(i)}(\underline{X}) = -\log p\ (\underline{X}/m_i) \qquad \ldots. \ (24)$$

Equation (23) can be written as

$$\tilde{\bar{n}}_N^{(i)} = \sum_{\underline{X}} \sum_{S \in W_i} p\,(\underline{X}/S,\ \underline{m}_i)\ p\,(\underline{S}/\underline{m}_i)\ n^{(i)}(\underline{X})$$

$$= \sum_{\underline{X}} \sum_{\underline{S} \in W_i} p\,(\underline{X}/\underline{m}_i)\ p\,(\underline{S}/\underline{m}_i)\ n^{(i)}(\underline{X})$$

$$= \sum_{\underline{X}} \sum_{\substack{\underline{S} \in W \\ \underline{S} \neq \underline{V}_i}} p\,(\underline{X}/\underline{m}_i)\ p\,(\underline{S}/\underline{m}_i)\ n^{(i)}(\underline{X}) + \sum_{\underline{X}} p\,(X/S=m_i)\ p\,(S=V_i/W_i)\ n^{(i)}(\underline{X})$$

or

$$\tilde{\bar{n}}_N^{(i)} = \bar{\Delta}^{(i)} + \bar{n}_N^{(i)} \qquad \text{.... (25)}$$

where

$$\bar{\Delta}^{(i)} = \sum_{\underline{X}} n^{(i)}(\underline{X})\ p\,(\underline{X}/\underline{m}_i) \sum_{\substack{\underline{S}\ \mathrm{E}\ W_i \\ \underline{S} \neq \underline{V}}} p(\underline{S}/\underline{m}_i)$$

$$= \sum_{\underline{X}} n^{(i)}(\underline{X})\ p\,(\underline{X}/\underline{m}_i)\ p\,(e/\underline{m}_i) \qquad \text{.... (26)}$$

or

$$\bar{\Delta}^{(i)} = p(e/m_i)\ \bar{n}_N^{(i)} \qquad \text{.... (27)}$$

Thus, the average code word length for the scheme in Fig. 3 is

$$\tilde{\bar{n}}_N = \sum_{i=1}^{K} p(m_i)\ \tilde{\bar{n}}^{(i)}$$

$$= \sum_{i=1}^{K} p(m_i)\ \bar{N}^{(i)} + \bar{\Delta} \qquad \text{.... (28)}$$

where

$$\bar{\Delta} = \sum_{i=1}^{K} p(e/m_i)\ \bar{n}_N^{(i)}\ .\ p(m_i) \qquad \text{.... (29)}$$

$$\bar{\Delta} \leqq p(e)\ .\ \max_{1 \leq i \leq K} \bar{n}_N^{(i)} \qquad \text{.... (30)}$$

or

$$\bar{\Delta} \leq D\ .\ \max_{1 \leq i \leq K} \bar{n}_N^{(i)} \qquad \text{.... (31)}$$

which indicates the deviation from the ideal case is a function of the average distortion.

APPENDIX 1

From noiseless coding theorem, the average code word length of a uniquely decipherable code for the composite source with a mixture distribution

$$\tilde{p}_j = \sum_{i=1}^{h} q_i p_{ij} \quad , \ j = 1, \ldots, M \qquad \text{.... (3)}$$

is bounded by

$$\bar{\tilde{n}} \geq \frac{H(X)}{\log Q_c} \qquad \text{.... (I.1)}$$

where $H(X) = - \sum_{j=1}^{M} p_j \log p_j$

$$H(X) = - \sum_{j=1}^{M} \left(\sum_{i=1}^{h} q_i p_{ij} \right) \log \left(\sum_{i=1}^{h} q_i p_{ij} \right)$$

Since the entropy is strictly convex, then

$$H(X) < \sum_{i=1}^{h} q_i \left[- \sum_{j=1}^{M} p_{ij} \log p_{ij} \right] \qquad \text{.... (I.2)}$$

$$H(X) \leq \sum_{i=1}^{h} H(X/S=b_i) q_i = H(X/S) \qquad \text{.... (I.3)}$$

and Q_c is the size of the code alphabet

Thus, $\bar{\tilde{n}} > \dfrac{H(X/S)}{\log Q_c}$

APPENDIX II

Let C_i be the optimal variable length encoder for the subsource i, i.e., to each alphabet a_j, j=1,..., M of the subsource i we assign a code word of length $n_j^{(i)}$ satisfying

$$\frac{-\log p_{ij}}{\log Q_c} \leq n_j^{(i)} < \frac{-\log p_{ij}}{\log Q_c} + 1 \qquad i=1,\ldots,h \qquad \text{.... (II.1)}$$

Suppose to each alphabet of the composite source we assign a code word of length

$$\hat{n}_j = \sum_{i=1}^{h} n_j^{(i)} \Pr\{S=b_i/X=a_j\} \qquad j = 1,\ldots, M \qquad \text{.... (II.2)}$$

where

$$\Pr(S=b_i/S=a_j\} = \frac{q_i p_{ij}}{\sum_{i=1}^{h} q_i p_{ij}} \qquad \text{.... (II.3)}$$

then the average code word length for this scheme is

$$\bar{\hat{n}} = \sum_{j=1}^{M} \hat{n}_j \Pr\{X=a_i\} \qquad \text{.... (II.4)}$$

or

$$\bar{\hat{n}} = \sum_{j=1}^{M} \sum_{i=1}^{h} \Pr\{X=a_i\} \Pr\{S=b_i/X=a_j\} n_j^{(i)} \qquad \text{.... (II.5)}$$

Substitution of (II.1) in (II.5) yields

$$\frac{H(X/S)}{\log Q_c} \leq n < \frac{H(X/S)}{\log Q_c} + 1$$

REFERENCES

1. T. Berger, Rate distortion theory: a mathematical basis for data compression, Prentice Hall, New Jersey, 1971.
2. R. G. Gallager, Information theory and reliable communication, Wiley, New York, 1968.

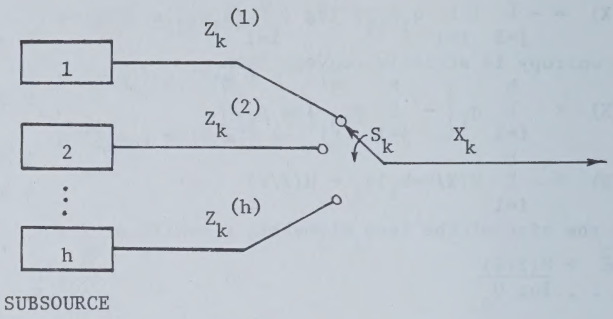

$Z_k^{(1)}$
$Z_k^{(2)}$
$Z_k^{(h)}$
1
2
h
S_k
X_k
SUBSOURCE

Fig. 1 Composite Source Model

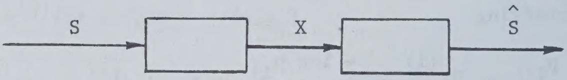


Fig. 2 Information Transfer in Cascaded Systems

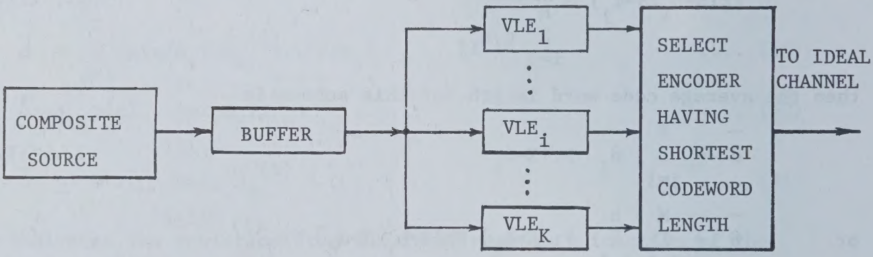

COMPOSITE SOURCE
BUFFER
VLE_1
VLE_i
VLE_K
SELECT ENCODER HAVING SHORTEST CODEWORD LENGTH
TO IDEAL CHANNEL

Fig. 3 Variable-Length Encoding

EVALUATING PERFORMANCE OF CODES FOR SPREAD SPECTRUM MULTIPLE ACCESS COMMUNICATIONS

MICHAEL B. PURSLEY
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois 61801

ABSTRACT

In certain communication systems it is desirable for many users to be able to transmit simultaneously over the same channel. This is the situation in many satellite communications systems, for instance, especially when the satellite has a single wide-band repeater. One method of permitting several users to transmit information simultaneously over a single channel is to phase modulate each carrier with a different binary code. Such a system is referred to in the literature as a phase-coded spread spectrum multiple access system or simply a spread spectrum multiple access (SSMA) system. In such a system the various users' signals may occupy the full channel bandwidth at all times during transmission and there will be mutual interference among these signals.

An analysis is presented which yields a method of quantitatively evaluating this interference in a SSMA system. The amount of interference depends on the type of codes employed and on the number of users. An expression for the signal-to-noise ratio as a function of code parameters is obtained. Numerical results are given for a SSMA system using Gold codes of period 511.

INTRODUCTION

The model that we will consider for a phase-coded SSMA system is shown in Figure 1. The signal $b_i(t)$ represents the binary data stream to be transmitted over the channel. The data signals are modulated onto a phase-coded carrier. The phase shifts induced by the code are $\pm\pi/2$ radians so that the phase-coded carrier can be viewed as a carrier which has been amplitude modulated by the code. The signal $a_i(t)$ represents the i^{th} code waveform.

Part of this work was done while the author was with the Hughes Aircraft Company, Los Angeles, California.

In a completely synchronous SSMA system, the various transmitted signals are assumed to have a common timing reference so that both the data and the code of a given signal are synchronized with all other signals. In addition the time delays τ_i and phase angles θ_i are assumed to be equal so that this synchronization is maintained by the modulation and transmission processes. For most applications it is not reasonable to require precise synchronization between the various transmitters. Consequently, we will consider an asynchronous SSMA system in which we allow the time delay's τ_i and phase angles θ_i to be random variables. In the most general model these variables would be random time functions representing, for example, oscillator drift or perhaps path length variations in a satellite system. Typically, the larger components of these random time functions will be slowing varying with respect to the data rate so that we can assume that they are approximately constant for the duration of a data symbol.

The receiver is presented a signal r(t) consisting of the sum of the K signals and noise from which it must extract and separate the relevant information for the K users of the system. In this paper we are concerned with performance measures which can be obtained for a given system with a reasonable amount of computational effort. Furthermore, we are interested in those SSMA systems where the transmission system requirements are determined by the data quality specifications (e.g., error rate or data signal-to-noise ratio). This would typically be the case for high data rate systems for instance. For a very low data rate, the code and carrier acquisition and synchronization specifications might actually determine the transmission system requirements. Even in this latter case, however, the analysis that we present will be useful in determining bounds on the system performance and in comparing various codes for SSMA applications.

ANALYSIS

The analytical model for a correlation receiver for signal 1 is shown in Figure 2. In general the received signal consists of K multiplexed signals corrupted by additive white Gaussian noise n(t) which has two-sided spectral density $N_o/2$ watts/Hz. Specifically,

$$r(t) = \sum_{i=1}^{K} \sqrt{2P}\, a_i(t-\tau_i) b_i(t-\tau_i) \cos(\omega_c t + \theta_i) + n(t) \qquad (1)$$

where we assume the random variables θ_i are independent and uniform on $[0,2\pi]$ and the random variables τ_i are independent and uniform on $[0,T]$ where T is the common data symbol duration for the K signals. For each i, $|a_i(t)| = |b_i(t)| = 1$ for all t so that the power in each of the K signals is P. For purposes of the present analysis we assume that the receiver is completely synchronized to the first signal and we are interested in the effect of the other K-1 signals on the quality of the first signal. Due to the large number of computations required in the numerical evaluation of the exact solution for the error probability we will compute the signal-to-noise ratio at the output of the receiver. For cases of interest ($K > 10$, $N > 200$) the signal-to-noise ratio is the most useful performance parameter that can be obtained with a reasonable amount of computational effort.

Since we assume the receiver is synchronized to the first signal, we can let $\theta_1 = 0$ and $\tau_1 = 0$. The independent random variables $\theta_2,\theta_3,\ldots,\theta_K$, $\tau_2,\tau_3,\ldots,\tau_K$ then represent differential phase shifts and time delays with respect to the first signal. The output of the multiplier of Figure 2 is

$$y(t) = \sum_{i=1}^{K}\sqrt{\tfrac{1}{2}P}\, a_i(t-\tau_i)b_i(t-\tau_i)a_1(t)[\cos\theta_i + \cos(2\omega_c t+\theta_i)] + n(t)a_1(t)\cos\omega_c t\,.$$

The frequency response of a realistic hardware implementation for the system of Figure 2 is such that the effect of the double frequency component in y(t) is negligible if $\omega_c \gg 1/T$. The significant input to the integrator is therefore

$$\hat{y}(t) = \sum_{i=1}^{K} \sqrt{\tfrac{1}{2}P}\, a_i(t-\tau_i)b_i(t-\tau_i)a(t)\cos\theta_i + n(t)a_1(t)\cos\omega_c t. \tag{2}$$

The corresponding output at $t = T$ is

$$Z(T) = \sqrt{\tfrac{1}{2}P}\sum_{i=1}^{K}\int_0^T a_i(t-\tau_i)b_i(t-\tau_i)a_1(t)\cos\theta_i dt + \int_0^T a_1(t)n(t)\cos\omega_c t \tag{3}$$

Writing equation (3) as desired signal plus interference plus noise we have

$$Z(T) = \sqrt{\tfrac{1}{2}P}\int_0^T a_1^2(t)b_1(t)dt + \sqrt{\tfrac{1}{2}P}\sum_{i=2}^{K}\int_0^T a_i(t-\tau_i)b_i(t-\tau_i)a_i(t)dt\cos\theta_i$$
$$+ \int_0^T a_1(t)n(t)\cos\omega_c t\, dt. \tag{4}$$

The data signals b_i are defined by

$$b_i(t) = \sum_{n=-\infty}^{\infty} b_{i,n} p_T(t-nT) \tag{5}$$

where we assume the $b_{i,n}$ are independent identically distributed random variables taking values +1 and -1 with equal probability and the function p_T is the rectangular pulse function defined by

$$p_T(\lambda) = \begin{cases} 1 & 0 \leq \lambda < T \\ 0 & \text{otherwise.} \end{cases}$$

The magnitude of the desired signal component of Z(T) is $|E\{Z(T)|b_{1,0}\}| = \sqrt{\tfrac{1}{2}P}\,T$ while the variance of Z(T) for a given $b_{1,0}$ is

$$\begin{aligned} \mathrm{Var}\{Z(T)|b_{1,0}\} &= \tfrac{1}{2}P \sum_{i=2}^{K} E\{[\int_0^T a_i(t-\tau_i)b_i(t-\tau_i)a_1(t)dt]^2 \cos^2\theta_i | b_{i,0}\} \\ &\quad + E\{[\int_0^T a_1(t)n(t)\cos\omega_c t\, dt]^2\} \\ &= \tfrac{1}{4}P \sum_{i=2}^{K} 1/T \int_0^T E\{[\int_0^T a_i(t-\tau)b_i(t-\tau)a_1(t)dt]^2 | b_{i,0}\}d\tau \\ &\quad + \int_0^T a_1^2(t)(N_o/2)\cos^2\omega_c t\, dt. \end{aligned} \tag{6}$$

The integral $\int_0^T a_i(t-\tau)b_i(t-\tau)a_1(t)dt$ represents the cross-correlation of the codes a_i and a_1 if $b_i(t-\tau)$ is of the same sign for all $t \in [0,T]$. However, according to (5), $b_i(t-\tau)$ will change sign with probability ½ in the interval $[0,T]$ so that in general we do not get full cross-correlations. In fact, by substituting from equation (5) into equation (6) we have

$$\begin{aligned} \mathrm{Var}\{Z(T)|b_{1,0}\} &= \tfrac{1}{4}P \sum_{i=2}^{K} \frac{1}{T}\int_0^T E\{[b_{i,-1}\int_0^{\tau} a_i(t-\tau)a_1(t)dt \\ &\quad + b_{i,0}\int_{\tau}^{T} a_i(t-\tau)a_1(t)dt]^2 | b_{i,0}\}d\tau + \frac{N_o}{2}\int_0^T a_1^2(t)\cos^2\omega_c t\, dt \end{aligned} \tag{7}$$

which can be expressed in terms of the partial cross-correlation functions R_{ik} and $\hat{R}_{ik}$ defined by

$$R_{ik}(\tau) = \int_0^\tau a_i(t-\tau)a_k(t)dt \tag{8}$$

$$\hat{R}_{ik}(\tau) = \int_\tau^T a_i(t-\tau)a_k(t)dt. \tag{9}$$

The code waveforms a_i are defined by

$$a_i(t) = \sum_{j=-\infty}^{\infty} a_j^{(i)} p_{T_c}(t-jT_c) \tag{10}$$

where $a^{(i)}$ is the code sequence for the i^{th} signal which satisfies $|a_j^{(i)}| = 1$ for each j, and $a_j^{(i)} = a_{j+p}^{(i)}$ for some p and any j (i.e., $a^{(i)}$ is a sequence of +1's and -1's which is periodic with period p) and T_c is the code symbol duration. For the present analysis we assume that period p divides $N = T/T_c$ so that there are an integral number of code periods in the information symbol interval $[0,T]$. Using (8), (9), and (10) we can write (7) as

$$\mathrm{Var}\{Z(T)|b_{1,0}\} = \frac{P}{4T} \sum_{i=2}^{K} \int_0^T E\{[b_{i,-1}R_{i1}(\tau) + b_{i,0}\hat{R}_{ik}(\tau)]^2 | b_{i,0}\}d\tau$$
$$+ \frac{N_o}{2}\left\{\frac{T}{2} + \frac{\sin 2\omega_c T}{2\omega_c}\right\}. \tag{11}$$

For $\omega_c \gg \frac{1}{T}$ we can neglect the double frequency term in (11) and, since $b_{i,-1}$ and $b_{i,0}$ are independent zero mean random variables with $|b_{i,n}| = 1$ with probability 1, then

$$\mathrm{Var}\{Z(T)|b_{1,0}\} = \frac{P}{4T} \sum_{i=2}^{K} \int_0^T R_{i1}^2(\tau) + \hat{R}_{i1}^2(\tau)d\tau + \frac{N_oT}{4} \tag{12}$$

$$= \frac{P}{4T} \sum_{i=2}^{K} \sum_{\ell=0}^{N-1} \int_{\ell T_c}^{(\ell+1)T_c} R_{i1}^2(\tau) + \hat{R}_{i1}^2(\tau)d\tau + \frac{N_oT}{4}. \tag{13}$$

For any nonnegative integer ℓ less than N we note that R_{i1} and $\hat{R}_{i1}$ are linear continuous functions on the interval $[\ell T_c, (\ell+1)T_c]$. Furthermore,

$$R_{i1}(\ell T_c) = \int_0^{\ell T_c} a_i(t-\ell T_c)a_1(t)dt = \sum_{j=0}^{\ell-1} \int_{jT_c}^{(j+1)T_c} a_i(t-\ell T_c)a_1(t)dt \tag{14}$$

$$\hat{R}_{i1}(\ell T_c) = \int_{\ell T_c}^{T} a_i(t-\ell T_c)a_1(t)dt = \sum_{j=\ell}^{N-1} \int_{jT_c}^{(j+1)T_c} a_i(t-\ell T_c)a_1(t)dt. \tag{15}$$

We define the discrete partial cross-correlation functions ρ_{i1} and $\hat{\rho}_{i1}$ by

$$\rho_{i1}(\ell) = \sum_{j=0}^{\ell-1} a_{j-\ell}^{(i)}\, a_j^{(1)}, \quad \hat{\rho}_{i1}(\ell) = \sum_{j=\ell}^{N-1} a_{j-\ell}^{(i)}\, a_j^{(1)} .$$

From (10) we see that $a_i(t)$ is constant on the interval $[jT_c, (\ell+1)T_c]$, so that (14) and (15) become

$$R_{i1}(\ell T_c) = \sum_{j=0}^{\ell-1} a_{j-\ell}^{(i)}\, a_j^{(1)} T_c = \rho_{i1}(\ell) T_c$$

$$\hat{R}_{i1}(\ell T_c) = \sum_{j=\ell}^{N-1} a_{j-\ell}^{(i)}\, a_j^{(1)} T_c = \hat{\rho}_{i1}(\ell) T_c .$$

Since R_{i1} and $\hat{R}_{i1}$ are linear on $[\ell T_c, (\ell+1)]$, then for $\tau \in [\ell T_c, (\ell+1)T_c]$

$$R_{i1}(\tau) = \rho_{i1}(\ell) T_c + [\rho_{i1}(\ell+1) - \rho_{i1}(\ell)](\tau - \ell T_c) \tag{16}$$

$$\hat{R}_{i1}(\tau) = \hat{\rho}_{i1}(\ell) T_c + [\rho_{i1}(\ell+1) - \rho_{i1}(\ell)](\tau - \ell T_c). \tag{17}$$

Therefore

$$\begin{aligned} \int_{\ell T_c}^{(\ell+1)T_c} R_{i1}^2(\tau) + \hat{R}_{i1}^2(\tau) d\tau &= \int_0^{T_c} R_{i1}^2(\lambda+\ell T_c) + \hat{R}^2(\lambda+\ell T_c) d\lambda \\ &= \frac{T_c^3}{3} \{\rho_{i1}^2(\ell) + \rho_{i1}(\ell)\rho_{i1}(\ell+1) + \rho_{i1}^2(\ell+1) + \hat{\rho}_{i1}^2(\ell) \\ &\quad + \hat{\rho}_{i1}(\ell)\hat{\rho}_{i1}(\ell+1) + \hat{\rho}_{i1}^2(\ell+1)\} \end{aligned} \tag{18}$$

and (13) implies

$$\mathrm{Var}\{Z(T)|b_{1,0}\} = \frac{PT_c^3}{12T} \sum_{i=2}^{K} r_{i1} + \frac{N_o T}{4} \tag{19}$$

where

$$\begin{aligned} r_{ij} = \sum_{\ell=0}^{N-1} \{&\rho_{ij}^2(\ell) + \rho_{ij}(\ell)\rho_{ij}(\ell+1) + \rho_{ij}^2(\ell+1) + \hat{\rho}_{ij}^2(\ell) \\ &+ \hat{\rho}_{ij}(\ell)\hat{\rho}_{ij}(\ell+1) + \hat{\rho}_{ij}^2(\ell+1)\}. \end{aligned} \tag{20}$$

If we define the signal-to-noise ratio (SNR) as

$$\mathrm{SNR} = |E\{Z(T)|b_{1,0}\}| / [\mathrm{Var}\{Z(T)|b_{1,0}\}]^{\frac{1}{2}}$$

then

$$\mathrm{SNR} = \{\frac{T_c^3}{6T^3} \sum_{i=2}^{K} r_{i1} + \frac{N_o}{2PT}\}^{-\frac{1}{2}}.$$

Since the energy per data bit E is PT and since $T = NT_c$,

$$\mathrm{SNR} = \{\frac{1}{6N^3} \sum_{i=2}^{K} r_{i1} + \frac{N_o}{2E}\}^{-\frac{1}{2}}. \tag{21}$$

Equation (21) gives the ratio of the signal voltage to the rms noise voltage at the output of the correlation receiver at the sampling instant in terms of the partial cross-correlation, the number N of code symbols per data symbol, and the bit energy-to-noise power density ratio E/N_o.

NUMERICAL RESULTS

The results of applying equation (21) to a SSMA system employing Gold codes [1] of period 511 with one code period per data symbol are presented in Figure 3 and Figure 4. In Figure 3, the quantity $20 \log_{10}(\mathrm{SNR})$ is shown as a function of the data symbol energy to noise power density ratio E/N_o for various numbers of users K. In Figure 4, the degradation $\delta = 20 \log_{10}\{(\mathrm{SNR})^{-1}\sqrt{2E/N_o}\}$ is given as a function of E/N_o for various values of K.

DISCUSSION

The analysis presented here provides a performance parameter which is useful for phase-coded SSMA trade-off studies and preliminary system analysis and design. Computation of this parameter for a given code requires the calculation of the partial cross-correlation functions which can be carried out on a digital computer since only mod 2 summations are needed. In contrast, many numerical integrations are required for calculation of the error probability for such a system. In practice we would perform a more detailed analysis, possibly including a simulation, before completing a final system design. This detailed analysis would include carrier and code synchronization, for instance.

We note that one of the main objectives of a preliminary analysis is to determine the number of users that can be accommodated in a given SSMA system. As illustrated in Figures 3 and 4, the performance measure gives an indication of the signal power required for various numbers of users. An even simpler measure suitable for this purpose can be obtained by applying a bound on the maximum partial cross-correlation recently obtained by Welch [2]. Application of this bound to equation (21) gives a somewhat pessimistic value for SNR for a given code but it does provide an easily computed performance parameter that is useful for estimating required power levels and code lengths for a given number of users.

ACKNOWLEDGMENT

The computer program to evaluate the partial cross-correlation functions for Gold codes was written by Diana Lange of the Hughes Aircraft Company.

REFERENCES

[1] R. Gold, "Optimal Binary Sequences for Spread Spectrum Multiplexing," IEEE Transactions on Information Theory (correspondence), Vol. IT-13, No. 4, pp. 619-621, October 1967.

[2] L. R. Welch, "Lower Bounds on the Maximum Cross Correlation of Signals," IEEE Transactions on Information Theory (correspondence), Vol. IT-20, pp. 397-399, May 1974.

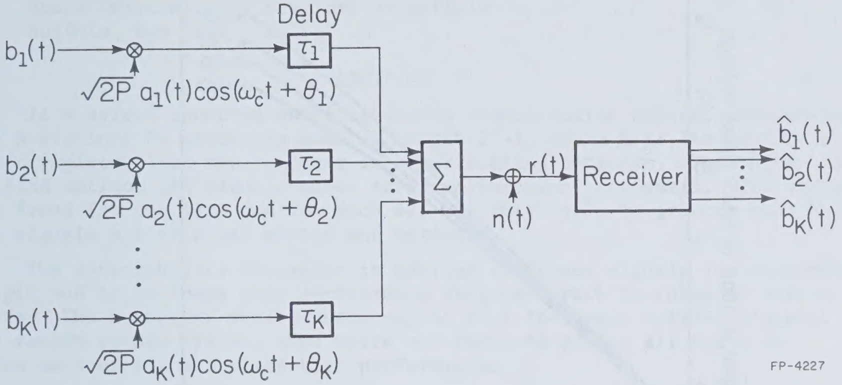

Delay
$b_1(t)$
$\sqrt{2P}\,a_1(t)\cos(\omega_c t + \theta_1)$
τ_1
$b_2(t)$
$\sqrt{2P}\,a_2(t)\cos(\omega_c t + \theta_2)$
τ_2
$b_K(t)$
$\sqrt{2P}\,a_K(t)\cos(\omega_c t + \theta_K)$
τ_K
Σ
$r(t)$
$n(t)$
Receiver
$\hat{b}_1(t)$
$\hat{b}_2(t)$
$\hat{b}_K(t)$
FP-4227

Figure 1. Phase-coded SSMA System model.

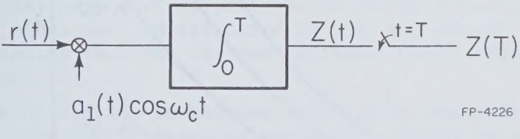

$r(t)$
$\int_0^T$
$Z(t)$
$t=T$
$Z(T)$
$a_1(t)\cos\omega_c t$
FP-4226

Figure 2. Analytical model for the synchronized correlation receiver.

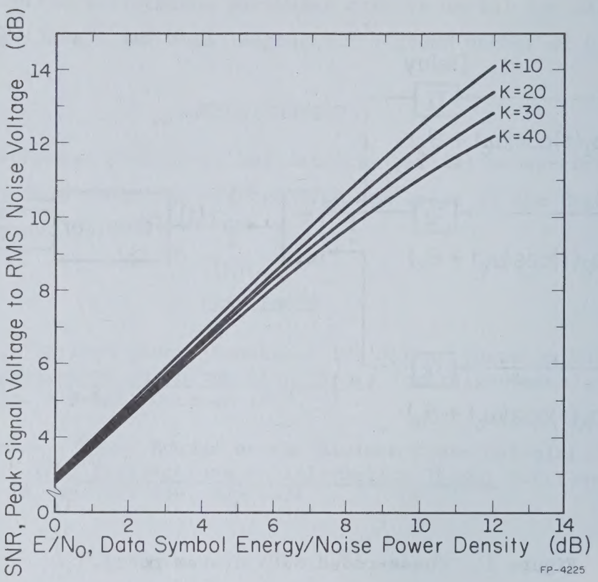


Figure 3. Performance of gold codes of period 511.

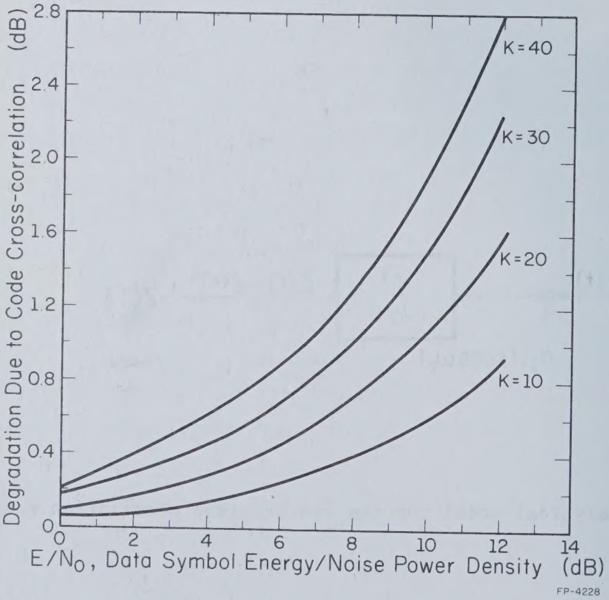


Figure 4. Performance of gold codes of period 511.

SIGNAL DESIGN FOR ASYNCHRONOUS MULTIPLE ACCESS COMMUNICATIONS

N. C. MOHANTY
Department of Electrical Engineering
State University of New York at Buffalo
Buffalo, New York 14214

ABSTRACT

In a spread spectrum multiple access communication system, transmitters use M distinct PN sequences each of length 2^N-1, where N is the number of shift registers and the receiver is a correlation detector. The object is to find optimum MPN signals whose cross correlations are small. The signals are found for two sets and for each set the duty cycle is greater than half. The signals are of equal energy and periodic.

The detectability parameter is plotted for these signals for different length and it is found that performance decreases with increase of active users. The number of active users may be 2^N-1 for these sets of signals. For random access system, when users are many and active all the time, these sets of signals yield high performance.

1. Introduction

The system in which a large number of users can be able to simultaneously interconnect their respective voice, data, teletype, facsimile and television communication links through a central facility without any dependence on any other users whatsoever with respect to frequency band occupancy or timing is called a multiple access system. Examples of such systems are multi-input analog and data processors and satellite networks. For the last fifteen years there have been intensive studies and experiments on satellite communications ([1],[2],[3]). The various aspects of launching, design and modulation technique of communication satellites are now well documented [4],[5],[6]. Communication satellites presently in orbit are capable of providing a communications capacity equal or superior to any other means available and through satellite repeaters it is possible to provide communication channels between widely separated earth stations [7]. It has been found useful to have traveling wave tube (TWT) amplifiers with their familiar hard limiting characteristics in satellite repeaters. Modulation techniques for various multiple access systems have been discussed by Aein [8] and Swartz, Aein and Kaiser [9]. For an asynchronous multiple access communication with a linear repeater, a coding technique, Employing orthogonal convolutional coding and Viterbi decoding, is described by Cohen et.al. [10]; they have derived performance characteristics that are for the case in which other user interference is the only source of noise. Channel capacity and coding theorems for the multiple access systems have been studied by Liao [11].

One of the modulation techniques with random and multiple access and which utilizes the entire repeater bandwidth, is the spread spectrum multiple access system (SSMA) with a hard limiter. These modulation techniques are similar to the code division multiple access system [12]. In SSMA, each user (earth station) is assigned a distinct pseudo noise carrier. The message is modulated into pseudo noise carrier and is transmitted through the satellite repeater. The receiver employs the correlation detector and then the receiver selects the message according to the output of the correlation detectors. The performance of this asynchronous system is given by its signal to noise ratio (SNR) detector output and this is called the detectability parameter. This parameter depends on input signal to noise ratio, number of users the repeater bandwidth and the maximum of the cross correlation output. In the present investigation, we seek optimal pseudo noise sequences which under a given input signal noise ratio and repeater bandwidth will enhance the performance measure. The pseudo noise signals will be generated using shift register sequences [13] and performance curves are plotted for these signals with respect to the number of users for different length of signals in the appendix. Maximum number of active users could be equal to the length of the signal.

System Model

There are M transmitter and each is assigned a signal

$$S_j(t) = \cos[\omega t + b_j(t-T_j)\Pi+\phi_j] \qquad 1 \leq j \leq M, \quad 0 \leq t \leq T$$

ω = carrier frequency common to all transmitter
Π = phase transition due to the synchronization sequence
ϕ_j = random phase, uniformly distributed independent random variables over (0,T)
T_j = the individual delays
$T = (2^N-1)\Delta T$
ΔT = chip duration
N - number of shift register

The all M sequences b_j $(1 \leq j \leq M)$ are of the same length, and all the transmitted signals are of the same energy and occupy the same bandwidth. Each transmitter contains a shift register generator and modulator. The M transmitted signals are summed into a hard limiter in the repeater. The system model is shown in Figure 1. The output of the repeater is amplified by a gain g and is exposed to the additive white Gaussian noise n(t) and is then passed through a bandpass filter. We assume no significant noise before the repeater and this is due to the fact that uplinks tend to have higher SNR than downlinks. The transmitters however need not be in chip synchronization or in word synchronization. The received signal is given by

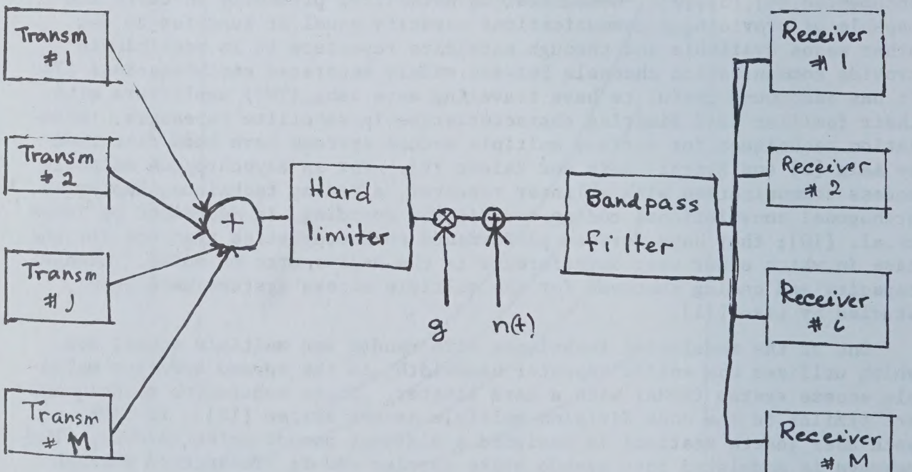


Figure L. System Model for Random Access

$$y(t) = g \text{ sgn } \Sigma\cos[\omega t + b_j (t-T_j)\Pi + \phi_j] + n(t)$$

where

$$\text{sgn } x = \begin{cases} 1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0 \end{cases}$$

The receiver is a correlation detector. The output of the i^{th} receiver whose reference signal $S_i(t)$ is given by

$$a_i = \frac{1}{T} \int_o^T y(t)S_i(t)di \quad 1 \leqslant i \leqslant M \tag{2.1}$$

This is defined in the sense of stochastic integral and α_i is not a Gaussian random variable. The receiver can detect a particular signal if the output of the correlation detector is equal to the energy of the signal. The cross-correlation of two PN_N sequences is given by

$$\rho_{ij} = \sum_{i=o}^{2^N-1} b_i b_{j+T} \tag{2.2}$$

Threshold is determined by

$$\rho = \max_{i,j} \rho_{ij}$$

Under certain assumptions([8],[9]), α_i and ρ_{ij} are equivalent and the detailed structure of transmitter and receiver are given by Schwartz et.al. [9]. The detectability parameter d is defined as

$$d^2 = \frac{(\bar{\alpha})^2}{\sigma_\alpha^2} \tag{2.3}$$

where

$$\bar{\alpha} = E\alpha_i \ Vi, \text{ (E is expectation operator)}$$

$$\sigma_\alpha^2 = E[\alpha_i - E\alpha_i]$$

By using the Fourier cosine series expansion ([14],[15])

$$\bar{\alpha} = E\alpha_i = \tfrac{1}{2}C_{1,00,0}$$

$$\sigma_\alpha^2 = \sigma_s^2 + \sigma_n^2 \tag{2.4}$$

But

$${}^{2}_{n} = \frac{N_0}{2(2^N-1)\Delta t}$$

Using the bounds [14]

$$\sigma_\alpha^2 = \Pi_c^2[\tfrac{1}{2}C^2_{0,1,00} + \tfrac{1}{2}C^2_{0,01,00} + \tfrac{1}{2}C^2_{0001}]$$

$$+\Pi_c^2 E \text{ [power in fundamental of intermodulation products]}$$

$$+ \frac{N_0}{2(2^N-1)\Delta T} \tag{2.5}$$

It is known that

$$C^2_{1,0,00} = C^2_{0,1,0,00} = \dots = C^2_{0,0\dots,1}$$

$$= \frac{1.76}{M} \text{ [statistical average of total receiver power in fundamental]}$$

and

$$E \begin{array}{l}\text{Power in fundamental of} \\ \text{modulation product}\end{array} = .12E[\text{received power in fundamental}] \qquad (2.6)$$

where Π_c is the peak of the in and out-of-phase cross correlation of MPN sequences. Therefore

$$d^2 \geq \frac{.44E(\text{received power at fundamental})}{\Pi_c^2[(M-1).88 + .12M]E(\text{received power}) + N_s} \qquad (2.7)$$

where

$$N_s = \frac{MN_0}{2(2^N-1)\Delta T}$$

Therefore for a given spectral density of noise and chip duration, the detectability parameter depends on M, the number of active users and the statistical average of required power in fundamental and cross correlation of M PM sequences.

1. Signal Design

In this context the objective is to design a maximum number of PN sequences (i.e. signals) whose cross correlation will be as small as possible so that the detectability parameter, the performance measure, can be enhanced. It is shown that [16] the ternary signal set [1,0,-1] is superior to binary maximal length sequences. Two sets of signals are derived as follows [16].

Type 1:

$$x_j^{\ell} = a_j d_{j+2\ell}(-1)^{a_{j+2}+b_{j+2}} \qquad 0 \leq \ell \leq 2^{N-2}$$

where

$$d_j = (-1)^{a_{j+1}+b_{j+5}} \qquad (0 \leq j \leq 2^N-1)$$

and a_j is a shift register sequence and $b_j = a_{3j}$ is a decimation.

These signals are periodic, equal energy, and orthogonal for some phase shifts. The duty cycle of this signal set is $\frac{1}{2}$. We can have more than 2^{N-2} with different initial conditions. M Could be equal to 2^N-1.

Let us define, periodic cross correlation as

$$C^N_{XY}(T) = \sum_{i=0}^{2^N-1} x_j y_{j+T}$$

and nonperiodic cross correlation as

$$D_{XY}(T) = \sum_{i=0}^{2^N-T-1} x_j y_{j+T}$$

where T is the delay or phase shifts and we define further

$$CC_N = \max_{\substack{X,Y \\ X \neq Y}} \left| C^N_{XY}(T) \right| \quad \forall T$$

$$DD_N = \max_{\substack{X,Y \\ X \neq Y}} \left| D^N_{XY}(T) \right| \quad \forall T$$

$$\Pi_c = \max\,[CC_N, DD_N]$$

For these sets of signals, N = 3, length of signals 7,

$$\Pi_c = 2 \qquad M \geqslant 3$$

and

N = 5, Length of signal 31

$$\Pi_c = 8$$

But there are signals whose cross correlation are as low as 4 for length of the signal 31, i.e.,

$$x^1 = [-1\ 0\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ -1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ -1\ 0\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0]$$

$$x^2 = [-1\ 0\ 0\ -1\ 0\ 1\ -1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ -1\ -1\ -1\ 1\ 0\ 1\ 0\ -1\ 0\ 0\ 0\ 0]$$

These two signals are also orthogonal and we can have at least 16 such signals.

For N - 7, Length is 127 and $M \geqslant 68$

$$\Pi_c = 16$$

and N = 9, length is 521 and $M \geqslant 256$

$$\Pi_c = 32$$

Type II:

$$y^{\ell}_j = \frac{1}{2} e_{j+2\ell} [(-1)^{a_j} + (-1)^{b_j}] \quad 0 < j < 2^{N-1}$$

where

$$c_j = (-1)^{a_{J+1} + b_{J+3}} \quad 0 \leqslant \ell \leqslant 2^{N-1}$$

These signals are periodic, equal energy and orthogonal for some phase shift. The duty cycle for this set of signals is greater than half. For these signals

N = 3, Length 7, $M \geqslant 4$, energy = 3

$$\Pi_c = 2$$

N = 5, Length 31, energy = 19, $M \geqslant 12$

$$\Pi_c = 7$$

N = 7, Length 127, energy = 71, $M \geqslant 36$

$$\Pi_c = 15$$

$N = 9$, Length 511, energy = 271, $M \geqslant 126$

$$\Pi_c = 33$$

A pair of signals for the second type with their correlation are given in the Appendix.

4. Performance and Conclusion

We have used a second set of signals to derive the detectability parameter. Though we have shown N to be odd prime, it could be other numbers as well. We plot the detectability parameter for the following parameters:

I. $N_0 = 1$, $\Delta T = 10^{-6}$ and $N = 3$

d^2 is plotted with respect to the number of users

II. $N = 5$, $N_0 = 1$, $\Delta T = 10^{-6}$

III. $N = 7$, $N_0 = 1$, $\Delta T = 10^{-6}$

Iv. $N = 9$, $N_0 =$; amd $\Delta T = 10^{-6}$

It is quite obvious from the formula for the detectability parameter that it increases whenever there are less users. Performance also increases with the increase of the size of the signal. It is shown [16] that the cross correlation decreases with increase of the signal length. These signals are designed such that individual delays need not be known to the receiver and we can have as many earth stations as 2^N-1. So there can be as many as 2^{N-1} active users. These signals are useful for small periodic as well as nonperiodic cross correlation. In addition to this, these signals have small periodic and nonperiodic autocorrelation.

References

[1] Proceedings of the IEEE, February, 1971, Special issue on Satellite Communication.

[2] Proceedings of IEE, IEE Review S. London, Vol. 119, No. 8R, August, 1972.

[3] IEEE Transaction on Communications, May, 1973, Special issue on Aeronautical Communications.

[4] Communication Satellite Systems Technology Progress in Astronautics and Aeronautics, Vol. 19, 1966, Academic Press, New York.

[5] Communication Satellites for the 70's: Technology, Feldman and Kelly, editors, Progress in Astronautics and Aeronautics, Vol. 25, 1971, Academic Press, New York.

[6] Communication Satellites for the 70's: System, Feldman and Kelly, editors, Progress in Astronautics and Aeronautics, Vol. 26, 1971, Academic Press, New York.

[7] E. Bedrosian, N. Feldman, G. Northrop and W. Sollfrey, "Multiple Access, Techniques for Communication Satellite: I Survey of Problem Rand RM-4298," September, 1964.

[8] J. M. Aein, "Multiple Access to a Hard-Limiting Communication-Satellite Repeater," IEEE Trans. on Space Electronics and Telemetry, December , 1964, pp. 159-167.

[9] J. W. Schwartz, J. M. Aein and J. Kaiser, "Modulation Techniques for Multiple Access to a Hard Limiting Satellite Repeater," Proc. IEEE, Vol. 5, No. 5, May, 1966, also see Proceedings of Multiple Access Summer Study, Inst. of Defense Analysis, Arlington, Va., Vol. I and II, IDA Report R-108, 1965.

[10] Andrew R. Cohen, J. A. Heller and A. J. Viterbi, "A New Coding Technique for Asynchronous Multiple Access Systems," IEEE Trans. on Communication Tech. Vol. COM-19, No. 5, Oct. 1971, pp. 849-855.

[11] Henry Hering-Jiunn Lio, "Multipole Access Channels," Ph.D. Dissertation, Dept. of Elec. Engr., University of Hawaii, Sept., 1972.

[12] Herman, Blasbalg, Hann F. Najjar, Renato A. D'Antonio and Raymond Haddad, "Air Ground Ground-Air Communications Using Pseudo-Noise Through a Satellite," IEEE Trans. on Aero. and Electronic Sys., Vol. AES-4, No. 5, September, 1964, pp. 774-791.

[13] S. W. Golomb, Shift Register Sequences, Holden-Day, Inc., San Francisco, Calif., 1967.

[14] Douglas R. Anderson and Paul A. Wintz, "Analysis of a Spread Spectrum Multipole Access System with a Hard Limiter," IEEE Trans. on Comm. Tech., Vol. COM-17, No. 2, April, 1969, pp. 285-290.

[15] Martin Nesenbergs and R. Godron Peterson, "Outputs to Signal Noise Ratio for a Random-Access Repeater Link with an Ideal Hard Limiter," IEEE Trans. on Aero. and Elec. Sys., Vol. AES-9, No. 3, May, 1973, pp. 448-459.

[16] N. C. Mohanty, "Signal Design for Small Correlation," Ph.D. Dissertation, Dept. of Elec. Engr., University of Southern California, February, 1973.

[17] N. C. Mohanty, "Polyphase Signals for Satellite Communication," International Conference on Communication, June, 1973.

[18] N. C. Mohanty, "Signals for Multiple Signal Environment," Electronic Science Laboratory, University of Southern California, Tech. 438, February, 1972.

COMBINATORIAL DECODING OF CERTAIN CODES

M. RAHMAN and I.F. BLAKE
Department of Electrical Engineering
University of Waterloo
Waterloo, Ontario, Canada

ABSTRACT

If the incidence matrix of a (v,b,r,k,λ) B I B D, is orthogonal to a code then it is possible to correct $\left\lfloor \frac{r+\lambda-1}{2\lambda} \right\rfloor$ errors with one-step majority logic decoding. This bound is generalised to the case when the incidence matrix is that of a t-design, for $t \geq 2$. Using this bound, the one-step majority logic decoding of the 1st, 2nd and 3rd order RM codes is examined.

1. INTRODUCTION

A t-design or a $\lambda_t;(t,k,v)$ tactical configuration is a collection of k-subsets (called blocks) from a set V of cardinality v, such that any t-subset appears in exactly λ_t blocks. If $\lambda_t=1$, the configuration is known as a Steiner System. It can easily be shown that every t-design is also a (t-1) design, where λ_{t-1}, is given by

$$\lambda_{t-1} = \frac{\lambda_t (v-t+1)}{(k-t+1)} \tag{1}$$

The derived tactical configuration, $\lambda_t;(t-1,\ k-1,\ v-1)$, is defined by the blocks of $\lambda_t;(t,k,v)$ containing any given element of V. The remaining blocks constitute the residual tactical configuration $\lambda_{t-1}-\lambda_t;\ (t,k,v-1)$.

Goethals[3] defined intersection numbers X_i^s to be the number of blocks, which intersect a given S-tuple in exactly i varieties. By counting all the distinct j-tuples in 2 different ways he obtained the following equation

$$\sum_{i=j}^{s} \binom{i}{j} X_i^s = \binom{s}{j} \lambda_j \tag{2}$$

Also, he pointed out that for $s \leq t$, there is a unique solution given by

$$X_j^s = \sum_{i=j}^{s} (-1)^{i-j} \binom{i}{j} \binom{s}{j} \lambda_i \tag{3}$$

Goethals[3] also defined the occurrence number of j-tuples Y_j^s, as the number blocks which intersect a given s-tuple in a given j-tuple. Clearly,

$$Y_j^s = \frac{X_j^s}{\binom{s}{j}} \tag{4}$$

2. The Generalised Lower Bound

Rudolph[1] showed that if the dual code supports a (v,b,r,k,λ) balanced incomplete block design, then $\left\lfloor \frac{r}{2\lambda} \right\rfloor$ errors can be corrected using 1-step majority logic decoding. This bound was modified to $\left\lfloor \frac{r+\lambda-1}{2\lambda} \right\rfloor$ by Ng[2] and Bastin[8], independently of each other. The purpose of this note is to modify this bound to the more general case, where the dual supports a t-design. The method used is an extension of Rudolph's arguments in the case of BIBD's.

Recently Goethals[3] and Assmuss and Mattson[5] independently discovered a one-step majority logic decoding algorithm for the extended binary (24,12) Golay code. The algorithm used the fact that the code was self-dual and the minimum wt. vectors supported the well known $1;(5,8,24)$ Steiner System. The problem with the algorithm is that the analysis is applicable only when $(t-1)$ or fewer errors are made, assuming the dual code supports a t-design. Unfortunately, no t-design for $t>5$ are known which implies a limitation on the application of the algorithm to larger codes with a distance greater than 10 or so. The purpose of this note however, is to examine the lower bound on the one-step majority logic error correcting capability of any binary code whose dual supports a t-design.

Let the dual code support a t-design and let H be the incidence matrix of the design. To decode the first bit, choose those rows of H which contain a '1' in the first position. There are exactly λ_1 such rows. The received sequence $\underline{r} = (r_o, r_1, r_2, \dots r_{n-1})$ is the vector sum of a code word $\underline{c}$ and an error vector $\underline{e}$. Consider the following λ_1 equations formed from the chosen λ_1 vectors of H.

$$\underline{r}.\underline{h}_i = 0 \qquad i = 1,2, \dots \lambda_1 \tag{5}$$

where $\underline{h}_i$ is a vector containing a '1' in the first position. By transposing the term containing r_o in (5), we have λ_1 estimates of r_o. A particular estimate of r_o will be correct if there are no errors affecting that equation and the estimate will be considered incorrect if there is any error affecting that equation. In addition to the λ_1 estimates of r_o, which are labelled $r_o^{(i)}$, $i=1,2,..\lambda_1$, we have the identity $r_o^{(o)} = r_o$, which we replicate A_k times, the meaning of A_k to be explained later. Thus there are λ_1+A_k estimates of r_o and if the total number of distinct estimates which are affected by k errors is less than $\frac{\lambda_1 + A_k - 1}{2}$, then a one-step majority logic algorithm guarantees the correction of k-errors or less.

The λ_1 rows which have been choosen from H, clearly yields the derived design λ_t; (t-1, k-1, v-1). Let the v-1 varieties be re-labelled 1,2, ... v-1. Let A_i be the number of those blocks containing the ith variety in the derived design, but not containing varieties i-1, i-2, ... 1. clearly,

$$A_i = Y_1^i \tag{6}$$

where Y_1^i refer to the occurrance number of i-tuples in the derived designs.

Lemma 1: for $i+1 \le t-1$, $A_i \ge A_{i+1}$

Proof: The proof is trival, as pointed out by Goethals [9].

$$A_i - A_{i+1} = Y_1^i - Y_1^{i+1} = Y_2^{i+1} \ge 0$$

Lemma 2: for any $i > t-1$, $A_i < A_{t-1}$

Proof: The proof follows from similar arguments.

Since the exact values of A_i, $i > t-1$ are not known, we assume the worst possible case and let $A_i = A_{t-1}$ for $i > t-1$. Let us suppose there are k errors. The total number of distinct estimates which are affected by k errors is $\sum_{i=1}^{k} A_i$. This is true, even if one error occurs in the first position, because we have A_k replications of the identity $r_o^{(o)} = r_o$. Thus by a simple majority vote, the code is capable of correcting k errors or less, if

$$\sum_{i=1}^{k} A_i \le \frac{\lambda_1 + A_{k-1}}{2} \tag{7}$$

for the particular case of a BIBD ie. a 2-design, $A_i = \lambda$, $\lambda_1 = r$ and equation (7) is equivalent to

$$k \le \left\lfloor \frac{r+\lambda-1}{2\lambda} \right\rfloor \tag{8}$$

which is the modified bound for Rudolph's algorithm.

3. Application

Lin[6] has shown that the class of Reed-Muller Codes are invariant under a triply-transitive permutation group. It is easy to check that the set of vectors which are invariant under a t-ply transitive group, support a t-design. A direct consequence of this is that all the vectors of a RM code, belonging to a fixed weight, form the incidence matrix of a t-design. The parameters of the 3-design obtained from the minimum weight vectors of a rth order RM Code are[6].

$$R = 2^{m-r} \quad ; \qquad v=n=2^m$$

$$\lambda_3 = \frac{(k-1)(k-2)}{(v-1)(v-2)} \prod_{i=o}^{m-r-1} \frac{2^m - 2^i}{2^{m-r}-2^i} \tag{9}$$

In order to decode an r-th order RM code, we have to look at the properties of the 3-design supported by the dual code, ie. the (m-r-1) the order RM code. Using equations (7) and (9), the following table was constructed, which shows the guaranteed 1-step majority logic decoding capability logic decoding capability (t). Also listed is the error-correcting capability (t*) using the modified Rudolph's algorithm[2].

n	r = 1		r = 2		r = 3	
	t	t*	t	t*	t	t*
16	3	2	1	1	-	-
32	5	5	3	2	1	1
64	11	10	5	4	2	2
128	22	21	9	9	5	4

It may be observed that in some cases, the new lower bound results in modest improvements. It is our conjecture that the improvements will be more pronounced if the parameter t, of the t-design, increases, ie. for t = 4 or 5.

4. Conclusions

An important theorem of Rudolph and Robbins[7] states that every linear code is 1-step weighted majority logic decodable. However, no systematic approach is known which yields the desired algorithm for any linear code. Looking at the t-design characteristics of the dual is one of the many possible approaches to the partial solution of the problem. It seems clear that vectors of more than one weight in the dual code will be required to achieve the full error - correcting capability of the code. On the other hand, vectors of different weight in a code, may lead to interesting, but less conventional combinatorial configurations.

REFERENCES

[1] L.D. Rudolph, A Class of Majority Logic Decodable Codes, IEEE Trans. Information Theory, IT-13, 305-307, (1967).

[2] S.W. Ng, On Rudolph's Majority Logic Decoding Algorithm, IEEE Trans. Information Theory, IT-16, 651-652, (1970).

[3] J.M. Goethals, On t-Designs and Threshold Decoding, Institute of Statistics Mimeo Series, No. 600.29, Univ. of North Carolina, Chapel Hill, June 1970.

[4] E.F. Assmus and H.F. Mattson, t-Design Decoding and the (48,24) Binary QR Code, G.T.E. Sylvania Report, AFCRL-71-0013, (1970).

[5] E.F. Assmus and H.F. Mattson, Majority Decoding of the (24,12) Binary Golay Code, G.T.E., Sylvania Report, Contract No. F-19628-69-C-0068, (1969).

[6] S. Lin, Some Codes which are Invariant under a Transitive Permutation Group and their Connection with Balanced Incomplete Block Designs, Proceedings of the Conference on Combinatorial Mathematics and its Application, Univ. of North Carolina, (1967).

[7] L.D. Rudolph and W.E. Robbins, One-step Weighted-Majority Decoding, IEEE Trans. Information Theory, IT-18, 446-448, (1972).

[8] G. Bastin, A propos du decodoge par decision majoritaire, Revue H.f. Vol. 8 (1970), 85-95.

[9] J.M. Goethals, Private Communication.

PROPERTIES OF GENERALIZED PLESS CODES

IAN F. BLAKE
Department of Electrical Engineering
University of Waterloo
Waterloo, Ontario, Canada

ABSTRACT

A class of square matrices which are orthogonal over the real numbers and contain only the elements 0, ± 1, is constructed. For certain parameters these matrices can be used to construct a class of self dual codes over GF(3) which can be shown to be a generalization of the class of symmetry codes.

I. INTRODUCTION

In a recent paper Pless [1] defined the class of symmetry codes over GF(3). They are self dual codes. Using a theorem of Assmus and Mattson [2], it can be shown that the incidence vectors of codewords of certain constant weights support 5-designs which were previously unknown. In this note a natural generalization of the Pless symmetry codes is given.

2. A CONSTRUCTION OF ORTHOGONAL MATRICES

Let G be a $k\times(q^k-1)$ matrix whose columns contain all the distinct nonzero k-tuples over the finite field GF(q). Using G we will construct a matrix $S(q^{k-1})$ containing only the elements 0, ± 1 and which will be orthogonal over the real numbers. The row space of the matrix G over GF(q) is equivalent to a maximum length code. In such a code it is known that every nonzero codeword has weight $(q-1)q^{k-1}$. Furthermore it is known (eg see [3]) that if G' is a $k\times(q^k-1)$ matrix whose rows are any set of k linearly independent vectors in the row space of G, then every nonzero k-tuple over GF(q) appears exactly once as a column of G'.

Let $n = (q^k-1)/(q-1)$ and H be a $k\times n$ submatrix of G with the property that any two columns of H are linearly independent. Assume that H is normalized so that the first nonzero element in each column is unity. Denote by A an $n\times n$ matrix whose rows are chosen from the row space of H with the property that any two rows of H are linearly independent, and assume that the first k rows are rows of H. Every row of H has weight q^{k-1}.

Let x_1 and x_2 be two distinct rows of A. Since they are independent we can extend them to a basis x_i, i=1, ..., k, each vector of which is a row of A, and let B be the $k\times n$ matrix with x_i as its i^{th} row, i=1, ..., k. Again, we assume that B has been normalized so that the first nonzero element in each column is unity. Let B' be the $k\times q^{k-1}$ submatrix of B consisting of those columns with unity in the first row. If B'' is the matrix B' with its first row removed, then each of the q^{k-1} (k-1) tuples over GF(q) appears exactly once as a column of B''. Each element of GF(q) then appears exactly q^{k-2} times in each row of B''.

Define a function

$$\chi: \; GF(q) \longrightarrow \{0, 1, -1\}$$

so that $\chi(0) = 0$, $\chi(\alpha) = +1$ if α is a square in GF(q) and -1 otherwise. We form the matrix $S(q^{k-1})$ from A by replacing each element α in A by $\chi(\alpha)$. Since each row of A has weight q^{k-1} and each nonzero element of GF(q) is either a square or nonsquare, the inner product of each row of $S(q^{k-1})$ with itself is q^{k-1} over the reals. Let $x_1 = (\alpha_1, \ldots, \alpha_n)$ and $x_2 = (\beta_1, \ldots, \beta_n)$ be two distinct rows of A and y_1, y_2 the corresponding rows of $S(q^{k-1})$. The inner product of y_1 and y_2 over the reals is the number of nonzero co-ordinate positions for which $\chi(\alpha_i) = \chi(\beta_i)$ less the number for which $\chi(\alpha_i) \neq \chi(\beta_i)$. Since χ is a multiplicative function, multiplication of a column of A by any element does not change the inner product of any two rows in $S(q^{k-1})$. From the comments on x_1 and x_2 made previously, the number of nonzero co-ordinate positions for which $\chi(\alpha_i) = \chi(\beta_i)$ is exactly equal to the number for which $\chi(\alpha_i) \neq \chi(\beta_i)$. Thus the inner product of y_1 and y_2 over the reals, is zero. We have shown:

Theorem. Over the real numbers

$$S(q^{k-1})\, S(q^{k-1})^T = q^{k-1} I_n$$

3. GENERALIZED SYMMETRY CODES

Suppose q and k are such that $q^{k-1} \equiv -1$ (3). In this case

$$S(q^{k-1})\, S(q^{k-1})^T = -I$$

over GF(3) and, consequently the matrix

$$G = [I_n : S(q^{k-1})] \tag{1}$$

generates a self dual code $C(q^{k-1})$ over GF(3). This occurs iff $q \equiv -1(3)$ and k is even. It can be shown that when k=2, the codes generated are equivalent to those of Pless. Thus $C(q^{k-1})$ is indeed a generalization of the Pless codes. Many of the properties of Pless codes are true also for the generalization. The following lemma is the image of lemma 3.1 in Pless [1].

Lemma. a). The weight of every row of G(eq(1)) is $q^{k-1} + 1$

b). The weight of any linear combination of two rows of G is $2q^{k-2} + q^{k-2} + (q-1)/2 + 2$.

c). The weight of any linear combination of three rows of G is at least

$q^{k-2}(q-3)/2 \quad + 5.$

4. COMMENTS

It would be interesting to know if the generalized codes presented here support 5-designs.

REFERENCES

1. V. Pless, Symmetry Codes over GF(3) and New Five Designs, J. Combinatorial Theory, 12, 119-142(1972).
2. E.F. Assmus Jr. and H.F. Mattson Jr., New 5-Designs, J. Combinatorial Theory, 6, 122-151, (1969).
3. J.J. Stiffler, "Theory of Synchronous Communications", Prentice-Hall, Englewood Cliffs, N.J., 1971.

A DIAGONALIZATION TECHNIQUE FOR SINGULAR PERTURBATION PROBLEMS

K.W. CHANG
Department of Mathematics, Statistics & Computing Science
The University of Calgary
Calgary, Alberta, Canada

ABSTRACT

A diagonalization technique is described for a singularly perturbed system of differential equations. Some applications are given.

1. TRIANGULAR FORM

When one encounters a nonlinear system of singular perturbation type

$$\begin{aligned} y' &= f(t,\varepsilon,y,z) \\ \varepsilon z' &= g(t,\varepsilon,y,z) \end{aligned} \tag{1}$$

it is natural to ask if the system (1) can be reduced to a simpler form (such as a block triangular form or a diagonal form), thereby rendering it more tractable. This method of approach has been employed in one form or another in several papers on singular perturbation problems. For example, in discussing existence of periodic solutions for the system (1), Flatto and Levinson [1] made the following assumption:

(H) The matrix function $U(t) = g_z^{-1}(t)g_y(t)$ has a continuous (and bounded) derivative on a t-interval R. Here $g_y(t)$ denotes the Jacobian matrix $g_y[t,0,\bar{y}(t),\bar{z}(t)]$, where $\bar{y}(t),\bar{z}(t)$ is the solution of the degenerate system obtained by formally setting $\varepsilon = 0$ in (1).

The hypothesis (H) is required to enable one to make the change of variables

$$v = y - \bar{y}(t), \quad w = z - \bar{z}(t) + U(t)[y - \bar{y}(t)] \tag{2}$$

in order to transform the system (1) into the "triangular form"

$$\begin{aligned} v' &= A_1(t)v + A_2(t)w + F(t,\varepsilon,v,w) \\ \varepsilon w' &= B(t)w \qquad\qquad + G(t,\varepsilon,v,w) \end{aligned} \tag{3}$$

where

$$\begin{aligned} A_1(t) &= f_y(t) - f_z(t)U(t), \\ A_2(t) &= f_z(t), \\ B(t) &= g_z(t), \end{aligned}$$

and

$$F = f(t,\varepsilon,\bar{y}+v,\bar{z}+w-Uv) - f(t,0,\bar{y},\bar{z}) - A_1(t)v - A_2(t)w$$

and G is defined analogously. If we assume that the functions f,g and their partial derivatives with respect to y,z are uniformly continuous and bounded, then the function F is continuous in (t,ε,v,w) and such that

$$|F(t,\varepsilon,0,0)| \to 0 \text{ as } \varepsilon \to 0.$$

Moreover, to any $\delta > 0$ there is a corresponding $\gamma(\delta) > 0$ such that

$$|F(t,\varepsilon,v,w) - F(t,\varepsilon,\bar{v},\bar{w})| \le \delta(|v-\bar{v}| + |w-\bar{w}|)$$

for $t \in R$, if $|v|,|\bar{v}|,|w|,|\bar{w}|,\varepsilon \le \gamma(\delta)$. The function G possesses similar properties.

If the transformation (2) is used without the term $U(t)[y-\bar{y}(t)]$, then the original system (1) is reduced to

$$\begin{aligned} v' &= f_y(t)v + A_2(t)w + F_1(t,\varepsilon,v,w) \\ \varepsilon w' &= C(t)v + B(t)w + G_1(t,\varepsilon,v,w), \end{aligned} \tag{4}$$

where $C(t) = g_y(t)$. This system is less tractable than the triangular system (3) owing to the presence of the term $C(t)v$. The effect of the term $U(t)[y-\bar{y}(t)]$ in (2) is to remove $C(t)v$ from (4).

The hypothesis (H) seems to be a fundamental one, as it was imposed by later authors who discussed (1) for existence of almost periodic solutions [2],[3] and for initial value problems over a finite interval [4],[5],[6],[7] or over a semi-infinite interval [8],[9].

In the papers [1] to [9] cited above, the following crucial hypothesis is also assumed or implied: There exists $\mu > 0$ such that every eigenvalue of $g_z(t)$ has absolute value $\ge 2\mu$ for $t \in R$. Then it can be shown [10] that hypothesis (H) is not needed at all to reduce system (1) to system (3). In fact, for ε small enough there exists a bounded, continuously differentiable matrix function $T(t,\varepsilon)$ such that

$$\varepsilon T' = B(t)T - \varepsilon T f_y(t) + \varepsilon T A_2(t)T - C(t). \tag{5}$$

The change of variables

$$v = y - \bar{y}(t), \quad u = z - \bar{z}(t) + T(t,\varepsilon)v \tag{6}$$

then transforms (1) into the triangular system

$$\begin{aligned} v' &= A^*(t)v + A_2(t)u + F^*(t,\varepsilon,v,u) \\ \varepsilon u' &= B(t)u + G^*(t,\varepsilon,v,u) \end{aligned} \tag{7}$$

where

$$F^* = f(t,\varepsilon,\bar{y}+v,\bar{z}+u-Tv) - f(t,0,\bar{y},\bar{z}) - A^*v - A_2u$$
$$\mathrm{A}^*(\mathrm{t}) = \mathrm{F}_\mathrm{y}(\mathrm{t}) - \mathrm{F}_\mathrm{z}(\mathrm{t})\mathrm{T}(\mathrm{t},\varepsilon)$$
$$G^* = g(t,\varepsilon,\bar{y}+v,\bar{z}+u-Tv) - \varepsilon\bar{z}' + \varepsilon T'v + \varepsilon T\{A^*v + A_2u + F^*\} - Bu.$$

The system (7) has the same form as (3), since F^*,G^* satisfy conditions of the same type as F,G of (3).

There is a relation between $U(t)$ and $T(t,\varepsilon)$, namely

$$\lim_{\varepsilon\to 0} T(t,\varepsilon) = B^{-1}(t)C(t) = U(t).$$

This is not too difficult to obtain from the matrix Riccati equation (5).

2. DIAGONALIZATION

The triangular system (3) or (7), to which (1) is reduced, now becomes tractable for problems involving existence of periodic or a.p.

solutions, and initial value problems. However, the system (3) or (7) is still not tractable enough for other problems, particularly boundary value problems. One needs to reduce (3) or (7) further to a block diagonal form. To this end Wasow [11] and Sibuya [12],[13] have achieved some degree of success. Their technique, as described in Wasow's book [14], results in a block diagonal form with a 'local' rather than a 'global' feature. Also they assume that the functions f, g are holomorphic.

Without imposing further assumptions than those already stated earlier, we shall now show that system (7) can be reduced at once to the following diagonal system

$$\begin{aligned} x' &= A^*(t)x + [\tilde{F}(t,\varepsilon,x,u) + S(t,\varepsilon)\tilde{G}(t,\varepsilon,x,u)] \\ \varepsilon u' &= B(t)u + \tilde{G}(t,\varepsilon,x,u), \end{aligned} \tag{8}$$

where

$$\begin{aligned} \tilde{F}(t,\varepsilon,x,u) &= F^*(t,\varepsilon,x-\varepsilon S(t,\varepsilon)u,u), \\ \tilde{G}(t,\varepsilon,x,u) &= G^*(t,\varepsilon,x-\varepsilon S(t,\varepsilon)u,u). \end{aligned}$$

This is achieved by applying to (7) a further change of variable

$$x = v + \varepsilon S(t,\varepsilon)u \tag{6'}$$

where $S = S(t,\varepsilon)$ satisfies the linear equation

$$\varepsilon S' = \varepsilon[f_y(t) - A_2(t)T(t,\varepsilon)]S - SB(t) - A_2(t). \tag{5'}$$

It suffices to remark that there exists $\varepsilon_0 > 0$ such that (5') has a solution $S(t,\varepsilon)$ which is uniformly bounded for $0 \le t \le 1$ and $0 < \varepsilon \le \varepsilon_0$. Moreover,

$$\lim_{\varepsilon \to 0} S(t,\varepsilon) = -A_2(t)B^{-1}(t).$$

Lastly, the transformations (6),(6') which achieve the diagonalization of system (1) can be expressed compactly as

$$\begin{pmatrix} y - \bar{y}(t) \\ z - \bar{z}(t) \end{pmatrix} = \begin{pmatrix} I & -\varepsilon S(t,\varepsilon) \\ -T(t,\varepsilon) & I + \varepsilon T(t,\varepsilon)S(t,\varepsilon) \end{pmatrix} \begin{pmatrix} x \\ u \end{pmatrix} \equiv H(t,\varepsilon)\begin{pmatrix} x \\ u \end{pmatrix}. \tag{9}$$

3. APPLICATIONS

In conclusion, we briefly mention three problems which become solvable in a straightforward manner through the application of the transformation above.

(I) The linear boundary value problem

$$\begin{aligned} y' &= A(t)y + B(t)z, \\ \varepsilon z' &= C(t)y + D(t)z, \end{aligned} \tag{10}$$

$$M(\varepsilon)\begin{pmatrix} y(0,\varepsilon) \\ z(0,\varepsilon) \end{pmatrix} + N(\varepsilon)\begin{pmatrix} y(1,\varepsilon) \\ z(1,\varepsilon) \end{pmatrix} = \begin{pmatrix} c_1(\varepsilon) \\ c_2(\varepsilon) \end{pmatrix}. \tag{11}$$

Here y, c_1 and z, c_2 are respectively m-dimensional and n-dimensional vectors and the coefficient matrices are of appropriate orders. We assume

(i) A,B,C,D are continuous and all eigenvalues of D have nonzero real part for $0 \le t \le 1$.

Then the equations (10) can be reduced to a purely diagonal form

$$x' = (A - BT)x$$
$$\varepsilon u' = (D + \varepsilon TB)u$$

by using the transformation (9)

$$\begin{pmatrix} y \\ z \end{pmatrix} = H(t,\varepsilon)\begin{pmatrix} x \\ u \end{pmatrix}$$

where T,S are bounded solutions of

$$\varepsilon T' = DT - \varepsilon TA + \varepsilon TBT - C,$$
$$\varepsilon S' = \varepsilon[A - BT]S - S[D + \varepsilon TB] - B$$

respectively.

Similar boundary value problems involving powers of ε have been discussed by Harris [15] and O'Malley [16]. Recently, Harris [17] has remarked on how the above diagonalization technique can be applied to other problems.

(II) The boundary value problem for the nonlinear vector equation

$$\varepsilon y'' = F(t,\varepsilon,y,y'), \quad y(0,\varepsilon) = \alpha(\varepsilon), \quad y(1,\varepsilon) = \beta(\varepsilon)$$

can be analyzed by linearization about the degenerate solution and then by applying the diagonalization transformation. Under appropriate assumptions as in the scalar boundary problem [18], this vector boundary problem can be solved [19].

(III) Linear control systems with large and small time constants or with slow and fast modes can be expressed in the form

$$y' = Ay + Bz + C_1\xi$$
$$\varepsilon z' = Cy + Dz + C_2\xi$$

where y,z and ξ are m,n and r vectors, respectively. Under suitable assumptions, the system can be reduced to a diagonal form, whose controllability properties are the same as those of the original system. The details are given in [20] and will not be reproduced here.

REFERENCES

1. L. Flatto and N. Levinson, Periodic solutions of singularity perturbed systems, J. Rational Mech. Anal. 4 (1955), 943-950.
2. J.K. Hale and G. Seifert, Bounded and almost periodic solutions of singularly perturbed equations, J. Math. Anal. Appl. 3 (1961), 18-24.
3. K.W. Chang, Almost periodic solutions of singularly perturbed systems of differential equations, J. Diff. Eq. 4 (1968), 300-307.
4. J.J. Levin, Singular perturbations of nonlinear systems of differential equations related to conditional stability, Duke Math. J. 23 (1956), 609-620.
5. __________, The asymptotic behaviour of the stable initial manifolds of a system of nonlinear differential equations, Trans. Amer. Math. Soc. 85 (1957), 357-368.

6. K.W. Chang and W.A. Coppel, Singular perturbations of initial value problems over a finite interval, Arch. Rational Mech. Anal. 32 (1969), 269-280.
7. F. Hoppensteadt, Properties of solutions of ordinary differential equations with small parameters, Comm. Pure Appl. Math. 24 (1971), 807-840.
8. V.F. Butuzov, Asymptotic behaviour of solutions of differential equations with a small parameter in the derivative on a semi-infinite interval, Vestnik Moskov Univ. Ser. I Mat. Meh. No. 1 (1965), 16-25.
9. K.W. Chang, Two problems in singular perturbations of differential equations, J. Austral. Math. Soc. 10 (1969), 33-50.
10. K.W. Chang, Remarks on a certain hypothesis in singular perturbations, Proc. Amer. Math. Soc. 23 (1969), 41-45.
11. W. Wasow, Asymptotic decomposition of systems of linear differential equations having poles with respect to a parameter, Rend. Circ. Nat. Palermo (2) 13 (1964), 329-344.
12. Y. Sibuya, Sur réduction analytique d'un système d'equations différentielles ordinaires linéaires contenant un paramètre, J. Fac. Sci. Univ. Tokyo (1), 7 (1958), 527-540.
13. ________, Asymptotic solutions of initial value problems of ordinary differential equations with a small parameter in the derivative, I and II, Arch. Rational Mech. Anal. 14 (1963), 304-311; 15 (1964), 247-262.
14. W. Wasow, Asymptotic expansions for ordinary differential equations, Interscience Publishers, New York, 1965.
15. W.A. Harris, Singular perturbations of two point boundary problems, J. Math. Mech. 11 (1962), 371-382.
16. R.E. O'Malley, Boundary value problems for linear systems of ordinary differential equations involving many small parameters, ibid. 18 (1969), 835-855.
17. W.A. Harris, Singularly perturbed boundary value problems revisited, Symp. on Ordinary Differential Equations, Lecture Notes in Math. 312, Springer-Verlag, 1973.
18. A. Erdélyi, On a nonlinear boundary problem involving a small parameter, J. Austral. Math. Soc. 2 (1962), 425-439.
19. K.W. Chang, Diagonalization method for a vector boundary problem of singular perturbation type, J. Math. Anal. Appl. (1974), to appear.
20. P.V. Kokotović and A.H. Haddad, Controllability and time-optimal control of systems with slow and fast modes, 1974 Decision and Control Conference.

NUMERICAL INTEGRATION OF GENERAL STIFF INITIAL-VALUE PROBLEMS BY SINGULAR PERTURBATION METHODS

LEON LAPIDUS, Department of Chemical Engineering, Princeton University,
Princeton, New Jersey 08540, and
RICHARD C. AIKEN, Technische Chemisches Laboratorium, Eidgenössiche
Technische Hochschule, CH-8006 Zürich, Switzerland

ABSTRACT

There is an equivalence between singularly perturbed ordinary differential equations and those with "stiff" stability requirements for numerical integration. This equivalence is identified and exploited in obtaining first-order inner and outer terms for the numerical solution of the general stiff problem not necessarily in singularly perturbed form. The zeroth-order outer solution is identified as the so-called "pseudo-steady-state approximation" (pssa) to the model often used on singularly perturbed equations to allow analytical solution or to aid data fitting. The remaining inner terms allow an *a priori* estimate of the length of the boundary-layer and thus the region of applicability while the first-order outer terms permit an accuracy evaluation of the pssa.

INTRODUCTION

Many commonly occurring dynamic systems have widely separated time constants. These systems are often represented by sets of initial value ordinary differential equations which possess some variables that rapidly change during time intervals much smaller than the transient phenomenon of interest. These systems have "stiff" requirements for adequate numerical integration. The commonly available integration routines require inordinately small step sizes when applied to such systems.

Examples of stiff systems abound in engineering science. Control processes may be stiff as a result of fast electrical control circuits coupled with slow mechanical movements [11, 19]. Electrical circuits may contain within the network semiconductors with very small response times compared to slower resistive elements [5, 17]. Energy component contributions may be quite directionally dependent in rocket and aircraft flight [6, 12], exhibiting stiff characteristics. In chemical kinetics, the stiff variables correspond to intermediate species that quickly react or are present in low concentrations [8]. Enzyme kinetics [15], polymerization [3], and chain reactions [7] illustrate this feature.

The step size (h) limitations in most standard methods for stable integration of an n-dimensional system require $\max_j |h\lambda_j|$, j = 1,2,...,n, where λ_j are local eigenvalues, be bounded by a single small number, typically in the 1 to 10 range [14]. If any eigenvalue is large in absolute value, corresponding to a fast mode, severe restrictions are placed on the step size. Depending on the length of the solution interval of interest, this may demand a large amount of computation time. In addition to the requirement of stable integration is accurate integration. If the fast modes significantly contribute to the solution in a finite region, the accuracy of approximating this contribution numerically is poorer the larger $|h\lambda_j|$. Although these errors will decay with a stable method, they may destroy the reliability of most of the computed transient. Further, there are limits on how small h can be before round-off errors accumulate and render the calculation worthless.

Thus, although any contribution to the solution by the stiff variables occurs only during a small portion of the transient (the "boundary layer"), determined by $\max|\lambda_j|$, this eigenvalue will continue to fix the maximum allowable step-size throughout the domain of interest. This domain is usually determined by $\min|\lambda_j|$; if not, a short domain may make a seemingly stiff problem appear non-stiff; alternately, a long domain may make a seemingly non-stiff problem appear stiff [13].

Certain characteristics common to most stiff integration applications may be itemized. Often the stiff variables are identifiable from physical reasoning, if not from examination of the system equations. The variables of primary interest, however, are usually the non-stiff ones outside of the boundary layer. This sharp boundary layer transient predominantly occurs initially without recurrence. Rarely are stiff modes precise representations of phenomena, therefore concentration on highly stable integration with moderate accuracy is favored.

THE EQUIVALENCE OF STIFF AND SINGULAR PERTURBED EQUATIONS

If the stiff variables may be identified from the n set

$$\dot{\underset{\sim}{z}}(t) = F(\underset{\sim}{z}), \underset{\sim}{z}(0) \tag{1}$$

then this set may be written in terms of the m stiff ($\underset{\sim}{y}$) and (n-m) non-stiff ($\underset{\sim}{x}$) variables:

$$\begin{aligned} \dot{\underset{\sim}{x}}(t) &= \underset{\sim}{f}(\underset{\sim}{x},\underset{\sim}{y}),\ \underset{\sim}{x}(0) = \underset{\sim}{\zeta} \\ \dot{\underset{\sim}{y}}(t) &= \underset{\sim}{w}(\underset{\sim}{x},\underset{\sim}{y}),\ \underset{\sim}{y}(0) = \underset{\sim}{\eta} \end{aligned} \tag{2}$$

If an artificial, small parameter ε is now introduced to emphasize the stiff nature of $\underset{\sim}{y}$,

$$\varepsilon\dot{\underset{\sim}{y}}(t) = \varepsilon\underset{\sim}{w}(\underset{\sim}{x},\underset{\sim}{y}) \equiv \underset{\sim}{g}(\underset{\sim}{x},\underset{\sim}{y}) \quad .$$

Thus (2) becomes

$$\dot{\underset{\sim}{x}}(t) = \underset{\sim}{f}(\underset{\sim}{x},\underset{\sim}{y}),\ \underset{\sim}{x}(0) = \underset{\sim}{\zeta} \tag{3a}$$

$$\varepsilon\dot{\underset{\sim}{y}}(t) = \underset{\sim}{g}(\underset{\sim}{x},\underset{\sim}{y}),\ \underset{\sim}{y}(0) = \underset{\sim}{\eta} \tag{3b}$$

typical of singular perturned form.

Aiken and Lapidus [2] have applied recent results on composite expansions for singular perturbed sets of the form (3) to derive an algorithm for the general stiff set (2). This algorithm calculates the zeroth- (subscript 0) and first-order (subscript 1) inner (upper case) and outer (lower case) terms for the stiff and non-stiff variables:

$$x^*(t) = X_0(t/\varepsilon) + \varepsilon X_1(t/\varepsilon) + x_0(t) + \varepsilon x_1(t) \quad ,$$

$$y^*(t) = Y_0(t/\varepsilon) + \varepsilon Y_1(t/\varepsilon) + y_0(t) + \varepsilon y_1(t) \quad ,$$

where $\left[x^*(t), y^*(t)\right]$ are the approximations to the solutions $\left[x(t), y(t)\right]$. The authors have shown how each of these eight terms may be computed independent of the expansion parameter ε, and is thus applicable to the general stiff system where such an identification is not obvious.

The zeroth-order terms contribute more significantly than the first-order terms the stiffer the system (hence the smaller ε). These terms are particularly interesting because of their simplicity:

$$\underset{\sim}{X}_0 = 0 \tag{4a}$$

$$\underset{\sim}{Y}_0(t/\varepsilon) = \underset{\sim}{Y}_0(0)e^{\underset{\approx}{w}_{y}(\underset{\sim}{x}_0,\underset{\sim}{y}_0)t} \tag{4b}$$

$$\underset{\sim}{Y}_0(0) + \underset{\sim}{y}_0(0) = \underset{\sim}{\eta}$$

$$\dot{\underset{\sim}{x}}_0(t) = \underset{\sim}{f}(\underset{\sim}{x}_0,\underset{\sim}{y}_0),\ \underset{\sim}{x}_0(0) = \underset{\sim}{\zeta}$$

$$0 = \underset{\sim}{w}(\underset{\sim}{x}_0,\underset{\sim}{y}_0) \tag{5b}$$

Note that $y_0(0) \neq \eta$ and must be found from (5b) with $\underset{\sim}{x}_0(0) = \underset{\sim}{\zeta}$. Equations 5 must be solved simultaneously for $\left(\underset{\sim}{x}_0(t),\ \underset{\sim}{y}_0(t)\right)$ after which $\underset{\sim}{Y}_0(t/\varepsilon)$ may be calculated.

Of the remaining four terms, the first-order outer terms are most important for present purposes:

$$\varepsilon x_1(t) = [\alpha\varepsilon x_1(0) - \beta]e^{\alpha t} + \beta/\alpha \tag{6a}$$

$$\varepsilon x_1(0) = Y_0(0)[f(\zeta,\eta) - f(\zeta,y_0(0))]/[2w(\zeta,\eta)]$$

$$\varepsilon y_1(t) = \gamma\varepsilon x_1(t) + \gamma f(x_0,y_0)/w_y(x_0,y_0) \tag{6b}$$

where

$$\alpha = f_x(x_0,y_0) - \gamma f_y(x_0,y_0)$$

$$\beta = \gamma f_y(x_0,y_0)/w_y(x_0,y_0)$$

$$\gamma = w_x(x_0,y_0)/w_y(x_0,y_0)$$

Thus $\varepsilon x_1(t)$ and $\varepsilon y_1(t)$ may be directly computed from knowledge of Y_0, x_0, and y_0. Note that w_y may not be singular.

THE PSEUDO-STEADY-STATE APPROXIMATION

One of the most common model simplifications consists of omitting variable terms which are multiplied by small parameters; when these terms are the highest order derivatives (the singular perturbed case (3)), the simplification is termed the pseudo-steady-state approximation (pssa) and corresponds to the zeroth-order outer approximation (5):

$$\dot{\underset{\sim}{x}}(t) = \underset{\sim}{f}(\underset{\sim}{x},\underset{\sim}{y}) \tag{7a}$$

$$0 = \underset{\sim}{w}(\underset{\sim}{x},\underset{\sim}{y}) \tag{7b}$$

Although this has been recognized earlier (interpreted only for systems in singular perturbed form [4]), there is nevertheless confusion as to the nature of the approximation.

First of all, observe that (7) is applicable to the general stiff system, not necessarily in singular perturned form; thus the pssa may mechanically be arrived at by simply setting $\dot{\underset{\sim}{y}} = 0$. This should not be interpreted as constancy of y necessarily over any interval. Various investigators have used this, however, as a criterion for application of the pssa [8,9]. If the singular perturbing ε also multiplies at least one term other than the highest derivative (regular perturbing), a pssa could be arrived at mechanically by setting $\varepsilon = 0$; for moderately stiff problems this may result in a larger error than retaining these regular perturbing terms [16].

The main concern with the pssa is its validity over the solution region; the accuracy over this valid region must then be evaluated. Validity of the pssa requires convergence of the infinite power series outer expansion in ε for small values of this parameter; if the exact solution approaches the zeroth-order approximation for ε approaching zero (regular degeneracy [18]), it may also approach the pssa for larger ε. Regular degeneracy corresponds to asymptotic stability of (3b) written as

$$\frac{d\underset{\sim}{y}}{d\tau} = \underset{\sim}{g}(\underset{\sim}{\alpha},\underset{\sim}{y})$$

when $\tau \equiv t/\varepsilon$ and $\underset{\sim}{x}$ is replaced by some constant vector $\underset{\sim}{\alpha}$ at each instant. A sufficient condition for this, as per Liapunov's First Method, is for $\underset{\approx}{g}_y$ to be negative definite. Less restrictive methods are given in [10]. Note that a non-linear $\underset{\sim}{g}$ may have several $\underset{\sim}{y}$ roots; hopefully the initial conditions will be within the stable root's region of influence.

Given that the actual solution does approach the pssa, the speed of approach, i.e., the region of pssa applicability, must be determined. Here considerations will be given only to those problems with stiffness (rapid transient) occurring initially, omitting "turning points"($\underset{\sim}{g}_y = 0$). For such problems, it should be expected that the pssa might be applicable only outside of the boundary layer. Application of the pssa within the boundary layer can produce gross errors that render the remaining portion of the transient useless. A most interesting interpretation of this boundary layer region is contained within the perturbation analysis: this small initial region corresponds to influence of the inner expansion. A simple measure of this influence is given by the zeroth-order inner term of the stiff variable (4). This expression can provide an a priori estimate of the boundary layer, particularly accurate for $\underset{\sim}{w}$ linear in $\underset{\sim}{y}$; the functional dependence on $\underset{\sim}{x}$ is not critical as this non-stiff variable is relatively constant over the interval.

Finally, the pssa accuracy within the valid region should be examined. Again, this is facilitated by our perturbation algorithm: the next higher approximation. If the contribution of the first-order terms is negligible compared to the zeroth-order, there is an indication that the system is stiff enough for an adequate application of the pssa.

EXAMPLE

An example is recorded here to illustrate both the full first-order perturbation solution and the pssa with boundary layer and accuracy estimates. All computation was performed on the IBM 360/91 at the Princeton University Computer Center. No special effort was made toward achieving an efficient package. The Runge-Kutta-Gill fourth-order (RKG4) method was used to integrate the zeroth-order outer solution both in the perturbing algorithm and in the pssa.

This example features moderate size, moderate stiffness, highly nonlinear form with multiple steady states:

$$\begin{aligned}
\frac{dx_1}{dt} &= \Omega(y_4 - x_1) + \alpha\beta y_1 e^{-\gamma/x_1} \\
\frac{dx_2}{dt} &= \Omega(y_4 - x_2) + \alpha\beta y_2 e^{-\gamma/x_2} \\
\frac{dy_1}{dt} &= (y_3 - y_1) - \alpha y_1 e^{-\gamma/x_1} \\
\frac{dy_2}{dt} &= (y_3 - y_2) - \alpha y_2 e^{-\gamma/x_2} \\
\frac{dy_3}{dt} &= (1 - y_3) + \phi_1[(y_1 + y_2)/2 - y_3] \\
\frac{dy_4}{dt} &= (1 - y_4) + \phi_2[(x_1 + x_2)/2 - y_4]
\end{aligned} \tag{8}$$

where $\alpha = 1.15 \times 10^6$, $\beta = 2.82 \times 10^{-4}$, $\gamma = 18.3$, $\phi_1 = 2.78$, $\phi_2 = 4.63$, and $\Omega = 10^{-3}$.

The steady-states are:

x_1, x_2	y_1, y_2	y_3	y_4
1.035	0.9170	1.029	0.9370
1.162	0.6138	1.133	0.7160
1.364	0.1340	1.299	0.3631

This system arose from fixed-bed reactor modeling considerations in which it was obvious that the thermal capacitance of the catalyst particles would govern the dynamics [1]; hence x_1 and x_2 are the non-stiff variables, y_1 to y_4 the stiff variables. This is also suggested directly by the magnitudes of the parameters in (8). Tables I - II contain the individual zeroth and first-order inner and outer terms for one non-stiff and one stiff variable, x_1 and y_1, respectively, as computed by the perturbation algorithm ($x_1(0) = x_2(0) = 1.162$, $y_1(0) = y_2(0) = 0.1340$, $y_3(0) = 0.3631$, $y_4(0) = 1.299$). Also given in these tables is the "exact" solution (x and y) computed by Gear's method (Gear, 1971: generally accepted as the best optimized stiff package available) to allow judgment on the reliability of the results. This package automatically adjusts the step size (and the order) as the integration proceeds to achieve specified accuracy requirements, so that the solution is not known at precise values of the independent variable. Thus presented in the tables for the exact solution are only those digits known to be correct at the listed times.

TABLE I. FULL PERTURBATION SOLUTION FOR NON-STIFF VARIABLES

t	x	X_0	εX_1	x_0	εx_1	x^*
0.01	1.162	0.0	-0.00011	1.16200	0.00011	1.16200
5.00	1.162	0.0	0.000003	1.16196	-0.00009	1.16195
25.00	1.162	0.0	-0.0000001	1.16192	10^{-9}	1.16192
100.00	1.162	0.0	3×10^{-10}	1.16193	7×10^{-10}	1.16193
5.2×10^4	1.175	0.0	-10^{-13}	1.17329	1×10^{-6}	1.17329
8.4×10^4	1.25	0.0	0.0	1.24641	0.000003	1.24642
10^5	1.325	0.0	0.0	1.31618	0.000002	1.31618
1.11×10^5	1.347	0.0	0.0	1.34389	5×10^{-7}	1.34389

TABLE II. FULL PERTURBATION SOLUTION FOR STIFF VARIABLE

t	y	Y_0	εY_1	y_0	εy_1	y^*
0.01	0.1361	-0.6056140	-0.000302	0.742693	-0.000383	0.136393
5.00	0.50784	-0.120243	-0.000042	0.614020	0.000065	0.493799
25.00	0.61373	-0.000341	-0.000028	0.614133	0.000035	0.613797
100.00	0.61405	0.000027	-0.000028	0.614133	0.000035	0.614167
5.2×10^4	0.5724	0.0	0.0	0.577418	0.000038	0.577456
8.4×10^4	0.3279	0.0	0.0	0.352688	0.000034	0.352723
10^5	0.1865	0.0	0.0	0.200821	0.000010	0.200830
1.11×10^5	0.1550	0.0	0.0	0.156713	0.000004	0.158717

In Table I the inner terms for the non-stiff variable are seen to be negligible, as expected. In Table II, however, for the stiff variable, the inner terms are found to not be significant (less than 0.1% contribution) beyond a time of about 25.0, corresponding to the length of the boundary layer. If (4b) is used to predict the time for 0.1% decay ($Y_0(t)/Y_0(0) = 0.001$), the estimated effective boundary layer length is 23.8, in good agreement with the computed transient. Observe that the zeroth-order outer terms are by far the dominant variables throughout the entire solution domain (the steady-state is reached at $t \approx 200.00$). The first-order outer terms as computed by (6) give only a qualitative feel for the degree of stiffness, hence the accuracy, of the pssa (compared with their significant contribution for a mildly stiff problem [2]).

The pssa is particularly appealing in this example since a decomposition can be affected by virtue of w being linear in y: the six-dimensional problem can be reduced to two dimensions. From the boundary-layer estimate, it is known that the first step must be greater than 25. Step sizes of 1000 were used to produce Table III, showing accuracy for the non-stiff variables comparable to that of Table I. Computation time for the entire solution was about two seconds; RKG4 without the pssa would take over five orders of magnitude more computation time for the same accuracy.

TABLE III. PSSA FOR NON-STIFF VARIABLE

$t \times 10^{-3}$	Full $x_{1,2}$	PSSA $x_{1,2}$
1	1.162	1.16203
52	1.175	1.17314
84	1.25	1.24553
100	1.325	1.31548
111	1.347	1.34356
129	1.360	1.35961
137	1.362	1.36173
146	1.363	1.36284

CONCLUSIONS AND SIGNIFICANCE

Approximations to the stiff solution have been indicated as a more effective procedure than direct numerical approximation. A numerical scheme has been developed to calculate first-order inner/outer expansions in an artificially introduced parameter that need not be identified. Hence the algorithm is applicable to the general stiff system not necessarily in singularly perturbed form.

From this algorithm, the pseudo-steady-state approximation (pssa) to the general stiff model has been given firm theoretical basis. The mechanical procedure of arriving at the pssa by setting the time derivatives of the stiff variables equal to zero is recommended even for problems obviously in singularly perturbed form. A viable means of estimating the accuracy and region of applicability, thus the boundary layer, of the pssa has been detailed. If a system is sufficiently stiff to merit special integration methods, the pssa, when properly applied, appears to yield accuracy quite adequate for most applications; naturally the stiffer the system, the better the approximation.

REFERENCES

1. Aiken, R.C., "Numerical Solution of Stiff Ordinary Differential Equations", Ph.D. Dissertation, Princeton University, 1973.

2. Aiken, R.C. and Lapidus, L., "An Effective Numerical Integration Method for Typical Stiff Systems", accepted for publication in AIChE Journal.

3. Bisenberger, J.A., and Capinpin, R., "The Quasi-Stationary State - Approximation in Polymerization Kinetics", J. Appl. Polymer Sci. 16, 695 (1972).

4. Bowen, J.R., Acrivos, A., and Oppenheim, A.K., "Singular Perturbation Refinement to Quasi-Steady-State Approximation in Chemical Kinetics", Chem. Eng. Sci. 18, 177 (1963).

5. Brayton, R.K., Gustavson, F.G., and Liniger, W., "A Numerical Analysis of the Transient Behavior of a Transistor Circuit", IBM J. Research and Development 10, 292 (1966).

6. Calise, A.J., "On the Use of Singular Perturbation Methods in the Solution of Variational Problems", presented at the Joint Automatic Control Conference, June 1973.

7. Creighton, J., "The Role of Numerical Stability in the Computer Solution of Chemical Chain Reactions", Proc. Summer Simulation Conference, Boston, 1971.

8. Edelson, D., and Allara, D.L., "Parameterization of Complex Reaction Systems: Model Fitting vs. Fundamental Kinetics", AIChE J. 19, 638 (1973).

9. Gelina, R.J., "Stiff Systems of Kinetic Equations -- A Practitioner's View", J. Comp. Physics 9, 222 (1972).

10. Hoppensteadt, F., "Properties of Solutions of Ordinary Differential Equations wtih Small Parametrics", Comm. Pure and Appl. Math. 24, 807 (1971).

11. Jamshidi, M., "A Near-Optimum Controller for Cold-Rolling Mill", Int. J. Control 6, 1137 (1972).

12. Kelley, H.J., "Flight Path Optimization with Multiple Time Scales", J. Aircraft 8, 238 (1971).

13. Lapidus, L., and Aiken, R.C., "Numerical Integration of Stiff Dynamic Systems", J. Dynamic Systems, Measurement and Control (in press).

14. Lapidus, L., and Seinfeld, J.H., Numerical Solution of Ordinary Differential Equations, Academic Press, New York, 1971.

15. Offen, H.A., and Deysens, L.N.M., "An Extension of the Steady-State Approximation of Kinetics of Enzyme-Containing Systems", J. Theor. Biol. 39, 387 (1973).

16. Ray, W.H., "The Quasi-Steady-State Approximation in Continuous Stirred Tank Reactors", Comm. J. Chem. Eng. 47, 503 (1969).

17. Richard, S., Peschard, J., Hribault, R., and Aube, G., "On Line Reduction Method for Computer Analysis of High-Order Electrical Circuits with Widely Separated Time Constants", IEEE Trans. Cir. Theory CT-19, 664 (1972).

18. Wascow, W., Asymptotic Expansions for Ordinary Differential Equations, Interscience Publishers, John Wiley & Sons, New York, 1965.

19. Yackel, R.A., and Kokotovic, P.V., "A Boundary Layer Method for the Matrix Riccati Equation", IEEE Trans. Automatic Control AC-18, 17 (1973).

ACKNOWLEDGMENT

This work was supported in part by the U. S. Army Research Office, Grant DAHC04-74-G-0158.

FURTHER RESULTS ON SINGULAR PERTURBATIONS AND SINGULAR ARCS*

R. E. O'Malley, Jr.
Department of Mathematics
University of Arizona
Tucson, Arizona 85721

Antony Jameson
Courant Institute
New York University
New York, N. Y. 10012

ABSTRACT

An asymptotic solution is obtained for the linear regulator problem with "cheap control". The essential new element is the singular perturbation solution to the matrix Riccati equation.

INTRODUCTION AND OUTLINE OF RESULTS

Consider the optimal control problem where

$$x = Ax + Bu, \quad 0 \leq t \leq T, \quad x(0) \text{ prescribed}, \tag{1}$$

with the scalar performance index

$$J(\epsilon) = \frac{1}{2}\int_0^t [x'Qx + \epsilon^2 u'Ru]dt \tag{2}$$

is to be minimized for ϵ a small, positive parameter (control is "cheap"). Here, A, B, Q, and R are smooth functions of t, with $Q \geq 0$ and $R > 0$ being symmetric matrices. The state x is an n vector, while the control u is an r vector. Asymptotic solutions of this problem were previously given by O'Malley and Jameson ([4], [8], [9], [10]) using singular perturbation techniques for the appropriate canonical equations. We refer to those papers for discussion of practical control applications leading to problems like (1)-(2) and interpretation of the results, noting only that (1)-(2) is a singular problem when $\epsilon = 0$ (cf. Bryson and Ho [2] and Ho [3]). Here we shall obtain a matrix Riccati solution, which has independent significance due to its feedback interpretation, numerical implementation, and extendability to the infinite interval problem (cf. Jameson and O'Malley [5]).

We first transform the problem through a change of variables previously used by Moylan and Moore [7] for the singular problem. Defining

$$u_1 = \int_0^t u(s)ds \quad \text{and} \quad x_1 = x - Bu_1, \tag{3}$$

we have the higher dimensional regulator problem

$$\begin{pmatrix}\dot{x}_1\\ u_1\end{pmatrix} = \mathcal{A}\begin{pmatrix}x_1\\ u_1\end{pmatrix} + \mathcal{B}u, \quad \begin{pmatrix}x_1(0)\\ u_1(0)\end{pmatrix} = \begin{pmatrix}x(0)\\ 0\end{pmatrix}, \tag{4}$$

---

*This work was supported by the Office of Naval Research under contract number N00014-67-A-0209-0022 and by the U. S. Atomic Energy Commission under contract number AT(11-1)-3077.

$$J(\epsilon) = \frac{1}{2}\int_0^T [\begin{pmatrix} x_1 \\ u_1 \end{pmatrix}' \mathcal{Q} \begin{pmatrix} x_1 \\ u_1 \end{pmatrix} + \epsilon^2 u'Ru]dt \tag{5}$$

where

$$\mathcal{A} = \begin{pmatrix} A & B_1 \\ 0 & 0 \end{pmatrix}, \quad \mathcal{B} = \begin{pmatrix} 0 \\ I \end{pmatrix}, \quad \text{and} \quad \mathcal{Q} = \begin{pmatrix} Q & QB \\ B'Q & B'QB \end{pmatrix}$$

for $B_1 = AB - \dot{B}$.

Using the standard technique (cf. Kalman [6] and e.g., Anderson and Moore [1]) of a feedback law

$$u = -\frac{1}{\epsilon^2} R^{-1} \mathcal{B}' \hat{k} \begin{pmatrix} x_1 \\ u_1 \end{pmatrix}, \tag{6}$$

the symmetric, positive semi-definite matrix $\hat{k}$ must satisfy the terminal value problem

$$\frac{d\hat{k}}{dt} = -\hat{k}\mathcal{A} - \mathcal{A}'\hat{k} + \frac{1}{\epsilon^2}\hat{k}\mathcal{B}R^{-1}\mathcal{B}'\hat{k} - \mathcal{Q}, \quad \hat{k}(T) = 0 \tag{7}$$

and there remains the initial value problem

$$\frac{d}{dt}\begin{pmatrix} x_1 \\ u_1 \end{pmatrix} = [\mathcal{A} - \frac{1}{\epsilon^2}\mathcal{B}R^{-1}\mathcal{B}'\hat{k}]\begin{pmatrix} x_1 \\ u_1 \end{pmatrix}, \quad \begin{pmatrix} x_1(0) \\ u_1(0) \end{pmatrix} = \begin{pmatrix} x(0) \\ 0 \end{pmatrix}. \tag{8}$$

Solution of the Riccati equation (7) (as in the singular problem) will differ considerably in a hierarchy of cases. We shall restrict attention here to the simplest case when

$$B'QB > 0, \tag{9}$$

while further cases will be discussed elsewhere. Note that (9) implies that $\mathcal{Q} \geq 0$ since

$$Q - QB(B'QB)^{-1}B'Q = E'QE \geq 0 \tag{10}$$

for $E = I - B(B'QB)^{-1}B'Q$. Putting $\hat{k}$ into its 2×2 block form, equation (7) can be written as three coupled equations for the independent elements. Analyzing them, one finds that the unique solution to (7) is of the form

$$\hat{k} = \begin{pmatrix} k & kB \\ B'k & B'kB \end{pmatrix} \tag{11}$$

where the $n \times n$ symmetric matrix k is the positive semi-definite solution of the singularly perturbed Riccati equation

$$\epsilon^2 \frac{dk}{dt} = -\epsilon^2(kA + A'k + Q) + kBR^{-1}B'k, \quad k(T) = 0. \tag{12}$$

The equations for the other components of $\hat{k}$ follow from

$$\epsilon^2 \frac{d}{dt}(kB) = -\epsilon^2(kB_1 + A'kB + QB) + kBR^{-1}B'kB \tag{13}$$

and

$$\epsilon^2 B' \frac{d}{dt}(kB) = -\epsilon^2 (B'kB + B'A'kB + B'QB) + B'kBR^{-1}B'kB. \tag{14}$$

Our principal result will be the asymptotic solution of the Riccati equation (12). For $B'QB > 0$, we will obtain an asymptotic solution of the form

$$k(t,\epsilon) = K(t,\epsilon) + \epsilon \ell(\sigma,\epsilon) \tag{15}$$

where the outer solution K has an asymptotic expansion

$$K(t,\epsilon) \sim \sum_{j\geq 0} K_j(t)\epsilon^j \tag{16}$$

and the terminal boundary layer correction ℓ has an expansion

$$\ell(\sigma,\epsilon) \sim \sum_{j\geq 0} \ell_j(\sigma)\epsilon^j \tag{17}$$

whose terms decay to zero as the stretched variable

$$\sigma = \frac{T-t}{\epsilon} \tag{18}$$

tends to infinity. Such expansions have been obtained for other singularly perturbed control problems by Yackel and Kokotović [12] and O'Malley and Kung [11]. Their direct approach doesn't seem to work here, so a more roundabout procedure (as in [5]) will be employed.

Once the expansions (15) are determined, the state equations (8) will have the asymptotic solution

$$\begin{aligned} x_1(t,\epsilon) &= X_1(t,\epsilon) + \epsilon m_1(\tau,\epsilon) + \epsilon^2 n_1(\sigma,\epsilon) \\ u_1(t,\epsilon) &= U_1(t,\epsilon) + v_1(\tau,\epsilon) + \epsilon w_1(\sigma,\epsilon) \end{aligned} \tag{19}$$

consisting of an outer solution (X_1,U_1), a right boundary layer correction $(\epsilon^2 n_1(\sigma,\epsilon),\ \epsilon w_1(\sigma,\epsilon))$ which tends to zero as $\sigma \to \infty$, and a left boundary layer correction $(\epsilon m_1(\tau,\epsilon), v_1(\tau,\epsilon))$ which tends to zero as the (left) stretched variable

$$\tau = \frac{t}{\epsilon} \tag{20}$$

tends to infinity. The transformation (3) then implies that the optimal control u and the corresponding trajectory x will be asymptotically represented in the form

$$\begin{aligned} u(t,\epsilon) &= \dot{u}_1 = U(t,\epsilon) + \frac{1}{\epsilon}v(\tau,\epsilon) + w(\sigma,\epsilon) \\ x(t,\epsilon) &= x_1 + Bu_1 = X(t,\epsilon) + m(\tau,\epsilon) + \epsilon n(\sigma,\epsilon) \quad . \end{aligned} \tag{21}$$

We note, in particular, that the optimal control will be unbounded at $t = 0$ as $\epsilon \to 0$ and that the optimal cost $J^*(\epsilon)$ will have an asymptotic series expansion. Further, the limiting solution within $(0,T)$ will agree with the singular arc solution given in [3] and elsewhere (cf. [11]).

THE RICCATI GAIN

Since the outer solution K represents the gain k asymptotically for $t < T$ $(\sigma \sim \infty)$, K must satisfy equations (12)-(14). When $\epsilon = 0$, note that (12) implies $K_0BR^{-1}B'K_0 = 0$, so the singular arc condition (cf. [2])

$$B'K_0 = K_0B = 0 \tag{22}$$

follows. (When B is invertible, then, $K_0 = 0$). Since $B'QB > 0$, (14) implies that $B'KB = O(\epsilon)$ is positive definite and, by (13),

$$KB = \epsilon^2[KB_1 + QB + \frac{d}{dt}(KB) + A'KB](B'KB)^{-1}R. \tag{23}$$

Substituting back into (12), then,

$$\begin{aligned}\frac{dK}{dt} + KA + A'K + Q = {} & [KB_1 + QB + \frac{d}{dt}(KB) + A'KB] \\ & \cdot [B'QB + B'\frac{d}{dt}(KB) + B'KB_1 + B'A'KB]^{-1} \\ & \cdot [B_1'K + B'Q + \frac{d}{dt}(B'K) + B'KA].\end{aligned} \tag{24}$$

Setting $\epsilon = 0$, we finally have the parameter-free Riccati equation

$$\begin{aligned}\frac{dK_0}{dt} & + K_0(A - B_1(B'QB)^{-1}B'Q) + (A' - QB(B'QB)^{-1}B_1')K_0 \\ & - K_0B_1(B'QB)^{-1}B_1'K_0 + (Q - QB(B'QB)^{-1}B'Q) = 0, \quad K_0(T) = 0.\end{aligned} \tag{25}$$

By (10), a unique K_0 exists throughout $0 \leq t \leq T$ (cf. [1]). Moreover, (22) will hold since $\frac{d}{dt}(K_0B) = 0$ whenever $K_0B = 0$. We observe that (25) does not follow from setting $\epsilon = 0$ in (12); that only implies (22).

Higher order terms in the outer expansion K are solutions of nonhomogeneous linear equations obtained from equating coefficients of higher powers of ϵ in (12)-(14). Specifically, for K_1, (13) and (14) imply that $B'K_1B > 0$ satisfies

$$(B'K_1B)R^{-1}(B'K_1B) = B'QB \tag{26}$$

while

$$K_1B = (K_0B_1 + QB)(B'K_1B)^{-1}R. \tag{27}$$

Since $K_1(T)$ cannot then be zero, a boundary layer correction $\ell_0(\sigma)$ to K_1 is needed to satisfy $k(T,\epsilon) = 0$. Further, the coefficient of ϵ in (24) implies a problem of the form

$$\frac{dK_1}{dt} + K_1[A - B_1(B'QB)^{-1}(B'Q + B_1'K_0)] +$$

$$+ [A' - (QB + K_0B)(B'QB)^{-1}B_1']K_1 = \alpha_1, \quad K_1(T) = -\ell_0(0) \tag{28}$$

where α_1 is determined by K_0 and $\ell_0(0)$ is, as yet, unspecified. Since (27)-(28) is a linearized version of the problem for K_0, its solution is unique.

Because the outer solution K satisfies (12), the representation (15) implies that the boundary layer correction $\ell(\sigma,\epsilon)$ must satisfy

$$\frac{d\ell}{d\sigma} = -\frac{1}{\epsilon}(\ell BR^{-1}B'K + KBR^{-1}B'\ell) - \ell BR^{-1}B'\ell + \epsilon(\ell A + A'\ell) \tag{29}$$

for $\sigma \geq 0$ and it must tend to zero as $\sigma \to \infty$. Introducing $C(T,\epsilon) = \epsilon^{-1}R^{-1/2}B'KBR^{-1/2}$, $L(\sigma,\epsilon) = R^{-1/2}B'\ell BR^{-1/2}$, and $\mathcal{L}(\sigma,\epsilon) = \ell BR^{-1/2}$, and setting $\epsilon = 0$ in (29) implies

$$\frac{dL_0}{d\sigma} = -L_0C_0(T) - C_0(T)L_0 - L_0^2 \tag{30}$$

$$\frac{d\mathcal{L}_0}{d\sigma} = -\mathcal{L}_0(C_0(T) + L_0(\sigma)) + \mathcal{L}_0(0)L_0(\sigma) \tag{31}$$

and

$$\frac{d\ell_0}{d\sigma} = \mathcal{L}_0\mathcal{L}_0'(0) + \mathcal{L}_0(0)\mathcal{L}_0' - \mathcal{L}_0(\sigma)\mathcal{L}_0 \tag{32}$$

while the initial condition $\ell_0(0) = -K_1(T)$ is unknown but completely determines

$$L_0(0) = -C_0(T) < 0 \quad\text{and}\quad \mathcal{L}_0(0) = -K_1(T)B(T)R^{-1/2}(T). \tag{33}$$

Integrating (30) and (31), then, we have

$$\mathcal{L}_0(\sigma) = 2\mathcal{L}_0(0)(I + e^{2C_0(T)\sigma})^{-1}. \tag{34}$$

Since $\ell_0 \to 0$ as $\sigma \to \infty$, we have

$$\ell_0(\sigma) = -\int_\sigma^\infty \frac{d\ell_0}{d\sigma}(s)ds = -2\mathcal{L}_0(0)(I + e^{2C_0(T)\sigma})^{-1}C_0^{-1}(T)\mathcal{L}_0'(0). \tag{35}$$

In particular, note that this completely determines $K_1(T)$. Higher order terms follow as solutions of linear equations.

REFERENCES

1. B. D. O. Anderson and J. B. Moore, Linear Optimal Control, Prentice-Hall, Englewood Cliffs, 1971.

2. A. E. Bryson and Y. C. Ho, Applied Optimal Control, Blaisdell, Waltham, 1969.

3. Y. C. Ho, "Linear Stochastic Singular Control Problems", J. Optimization Theory Appl. 9 (1972), 24-31.

4. A. Jameson, "The Connection Between Singular Perturbations and Singular Arcs: Part 2: A Theory for the Linear Regulator", Proceedings, Eleventh Annual Allerton Conference, October 1973, 686-692.

5. A. Jameson and R. E. O'Malley, Jr., "Cheap Control of the Time-invariant Regulator", Applied Math. and Optimization, to appear.

6. R. E. Kalman, "Contributions to the Theory of Optimal Control", Bol. Soc. Mat. Mexicana 5 (1960), 102-119.

7. P. J. Moylan and J. B. Moore, "Generalizations of Singular Optimal Control Theory", Automatica 7 (1971), 591-598.

8. R. E. O'Malley, Jr., "Examples Illustrating the Connection Between Singular Perturbations and Singular Arcs", Proceedings Eleventh Annual Allerton Conference, October 1973, 678-685.

9. R. E. O'Malley, Jr., "The Singular Perturbation Approach to Singular Arcs", Proceedings, Int. Conf. on Differential Equations, USC, September 1974.

10. R. E. O'Malley, Jr. and A. Jameson, "Singular Perturbations and Singular Arcs I", to appear.

11. R. E. O'Malley, Jr. and C. F. Kung, "The Matrix Riccati Approach to a Singularly Perturbed Regulator Problem", J. Differential Equations 16 (1974), to appear.

12. R. A. Yackel and P. V. Kokotović, "A Boundary Layer Method for the Matrix Riccati Equation", IEEE Trans. Automatic Control 18 (1973), 17-24.

APPROXIMATION OF HIGH ORDER LINEAR SYSTEMS BY LOWER ORDER MODELS

J.M. SIRET, G. MICHAILESCO
Laboratoire de Signaux et Systèmes-Fontenay-aux-Roses-France

The use of orthogonal projection, previously proposed by D. Mitra, is a powerful way to obtain an equivalent lower order model of a multivariable asymptotically stable high order system. This communication makes some original contributions in the choice of the best projection subspace and in the synthesis of reduced models. These lead to a much improved method applicable for inputs other than impulses.

INTRODUCTION

Analysis, simulation or even application of some known result of optimal control theory applied to linear dynamic systems of large dimensions are too cumbersome and often impracticable. Thus it is of interest to deal with a model of lower dimension which gives a reasonable description of the original system.

The reduction by orthogonal projection in the state space was initially proposed by D. Mitra [1] [2]. It consists of finding a subspace so that the error between the real and projected phase trajectories is minimal, and then synthetizing the model of lower dimension belonging to that subspace. However this method is far from satisfactory, for it is limited to the case with impulses as inputs and quite often gives poor results.

The reduction method proposed here is also based on orthogonal projection in state space. However, two important modifications are introduced in the choice of the matrix characterizing the subspace. Moreover a new technique for the synthesis of the reduced model is given. Several examples show the interest of the proposed method.

DEFINITION OF THE BEST SUBSPACE

Problem statement.

For the linear dynamic system, assumed to be invariant and asymptotically stable and defined by

$$\begin{cases} \dot{X} = AX + BU \\ S = CX \end{cases} \qquad (1)$$

where X, U and S are vectors of dimensions respectively n, p and q, find a model of lower dimension described by

$$\begin{cases} \dot{\tilde{X}}_1 = F\tilde{X}_1 + GU \\ \tilde{S} = H\tilde{X}_1 \end{cases} \qquad (2)$$

where $\tilde{X}_1$, vector of dimension $m (n > m \geq q)$, is, for a certain class of inputs, a faithful approximation of some components of the state vector X, which are of physical significance.

Basis for the projection subspace.

The first step of the reduction method consists of finding a subspace E_m of the state space E_n so that the functionnal J of the error between the real phase trajectory $X(t)$ and its projection $X_p(t)$ is minimized,

$$J = \int_0^{t_1} \| X(t) - X_p(t) \|_E^2 \ dt \tag{3}$$

where $\| X(t) - X_p(t) \|_E$ represents the Euclidian norm.

As shown in [1] , J is minimal if the subspace E_m is the one generated by the m eigenvectors associated with the m largest eigenvalues of the symmetric matrix

$$W(t_1) = \int_0^{t_1} X(t) \, . \, X^T(t) \ dt \tag{4}$$

Obviously the quality of the reduction depends essentially upon the choice of the subspace E_m, and thus upon the matrix W used. In particular when the system has a non zero steady state $X(\infty)$, it is preferable to introduce another matrix $W_y(t_1)$ which only characterizes the transient response.

Reduction of the order of a system will also generally give a poorer approximation of the fast modes present in the original system. Thus it is better advisable not to take into account the beginning of the transient response but rather to choose a lower bound t_o in the integral defining W_y :

$$W_y(t_o, t_1) = \int_{t_o}^{t_1} (X(t) - X(\infty)) \, . \, (X(t) - X(\infty))^T dt \tag{5}$$

The choice of t_o will depend upon the criteria used for characterizing the reducibility of the system. We propose to choose the lower bound of integration t_o as the smallest time such that

$$R = \frac{\lambda_{y,m}}{\lambda_{y,m+1}} \geqq R_o \tag{6}$$

where the $\lambda_{y,i}$, eigenvalues of $W_y(t_o, t_1)$, are in order of decreasing magnitude. This criterium for reducibility can be justified by observing that a large value of R implies that the (n-m) smaller eigenvalues of W_y are negligeable with respect to the m largest ; and we know that the system would be strictly reducible if the (n-m) smallest eigenvalues were equal to zero [3].

SYNTHESIS OF THE REDUCED MODEL

The second step of the reduction method consists of synthetizing a reduced model. Let us assume that the inputs are either impulses or step functions. When the criterium of reducibility (6) is satisfied, the error between the phase trajectory and its projection is small and we may consider that the state of the system belongs effectively to E_m. Then :

$$V.Y \cong 0 \quad \text{with} \quad Y(t) = X(t) - X(\infty) \tag{7}$$

where V^T is the matrix whose columns are the eigenvectors of the subspace E_{n-m}, orthogonal to E_m.

If Y_1 is the state vector, whose m components are the state variables to be retained for the reduced model, the initial system can be written in the partitioned form :

$$\begin{bmatrix} \dot{Y}_1 \\ \dot{Y}_2 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} \tag{8}$$

$$Z = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix}$$

and (7)

$$V_1 Y_1 + V_2 Y_2 \cong 0 \tag{9}$$

or

$$Y_2 = - V_2^{-1} . V_1 . Y_1 \tag{10}$$

Taking into account (8), (10) and the later change of variable we get for the matrices F, G, H of the reduced model defined by (2) the following expressions :

$$F = A_{11} - A_{12} V_2^{-1} V_1 \tag{11a}$$

$$G = - F X_1(\infty) \tag{11b}$$

$$H = C_1 - C_2 V_2^{-1} V_1 \tag{11c}$$

Note that this reduced model has the same steady state as the original system. However when the input is a unit impulse, equation (11b) does not define G, so other hypotheses have to be made (for example same initial conditions).

APPLICATION

In this section we present two examples of reduction for monovariable systems, in order to show the validity of the method.
Example 1 (fig. 1) : for the third order system of transmittance

$$H(p) = \frac{50}{50 + 65\,p + 16\,p^2 + p^3}$$

subject to a unit impulse input, two second order reduced models were obtained : the first (b) is synthetised by the method of D. Mitra with the interval of integration $(0, \infty)$, the second (c) by the method presented in this paper with t_0 satisfaying (6).
Example 2 (fig. 2) : another third order system, taken from [4] was considered :

$$H(p) = \frac{2 + 6\,p + 8\,p^2}{(p+1)^2 \quad (p+2)}$$

with a unit step as input. The second order reduced model (c) shows again a better approximation for the response than the model (b) obtained by the moments matching method.

For this two examples (a) is the response of the original system. The results concerning the other retained state variable, which are not given here, are also satisfactory.

CONCLUSION

A reduced order model for the approximation of linear systems of large dimension can be found by looking at the best subspace of the state space, such that the error between the real and projected trajectories is minimal. Obviously the quality of the reduction is directly related to the choice of the fonctionnal characterizing the projection error. Thus by modifying a certain integration interval and by eliminating the steady state (when present) of the original system, it has been possible to improve greatly the procedure for obtaining the best subspace of a given order. Moreover a new and easily applicable technique for synthesis of the model has been given.

This technique leads to particularly suitable results for reduction of systems having non zero steady state.

REFERENCES

[1] D. MITRA. "W matrix and the geometry of model equivalence and reduction". Proc. IEE, Vol. 116, n° 6, June 1969.

[2] D. MITRA. "Analytical results on the use of reduced models in control of linear dynamical systems". Proc. IEE, Vol. 116, n° 8, August 1969.

[3] R.-E. KALMAN. "Mathematical description of linear dynamical systems". SIAM J. Contr., série A, Vol. 1, n° 2, 1963.

[4] S.C. CHUANG. "Application of continued fraction method for modelling transfer functions to give more accurate initial transient response". Imperial College of Science and Technology, London, publication 70/30, December 1970.

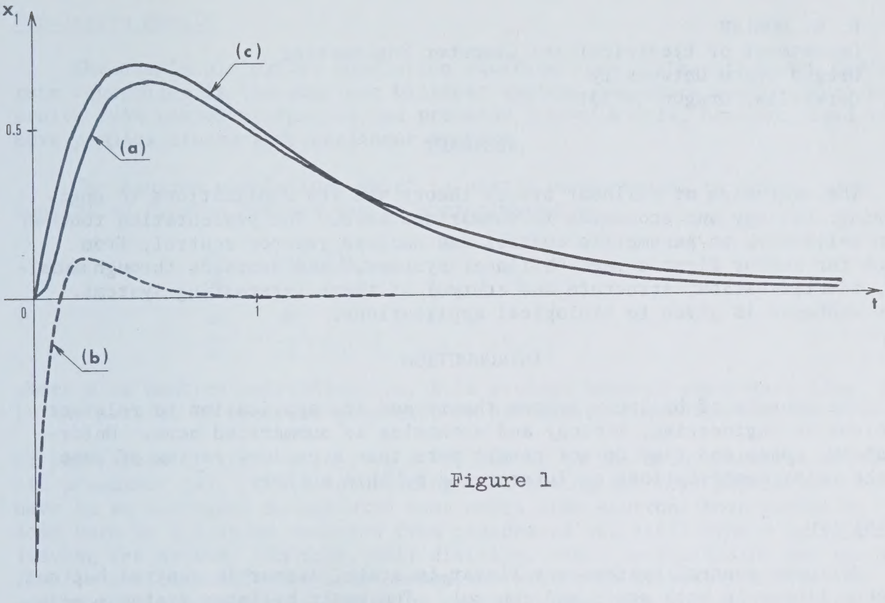


Figure 1

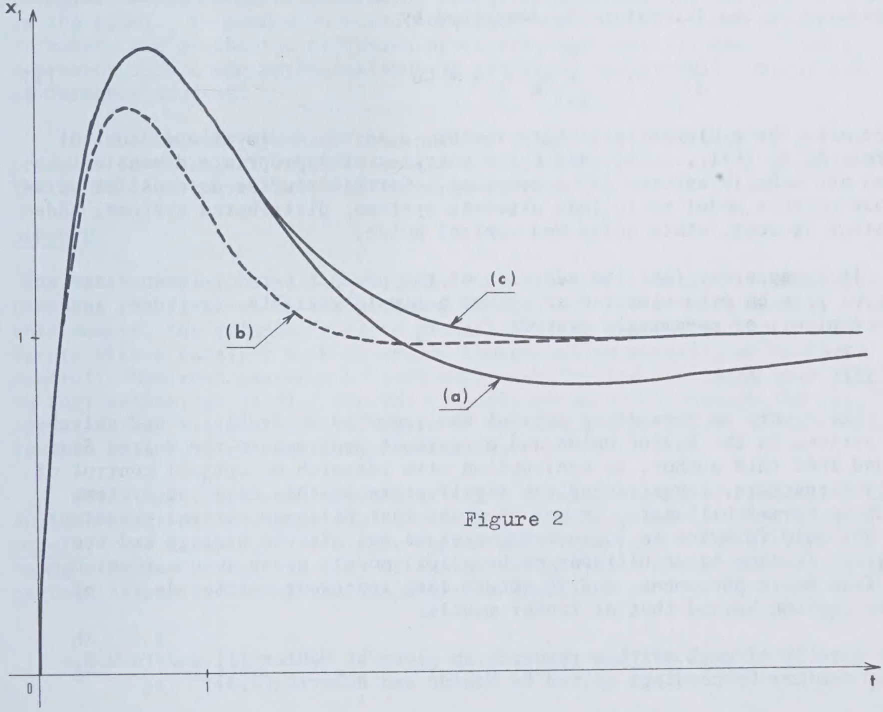


Figure 2

BILINEAR SYSTEMS: GENESIS AND DEVELOPMENT

R. R. MOHLER
Department of Electrical and Computer Engineering
Oregon State University
Corvallis, Oregon 97331

ABSTRACT

The evolution of bilinear system theory and its applications to engineering, biology and economics is summarized here. The presentation touches upon early work on parametric control and nuclear reactor control, from which the author first coined "bilinear systems," and proceeds through modeling, identification, structure and control of these interesting systems. Some emphasis is given to biological applications.

INTRODUCTION

The genesis of bilinear system theory and its application to relevant problems in engineering, biology and economics is summarized here. Unfortunately, space and time do not permit more than a cursory review of some of the major contributions as interpreted by this author.

Definition

Bilinear control systems are linear in state, linear in control but not jointly linear in both state and control. The basic bilinear system mostly considered in the literature is described by

$$\frac{dx}{dt} = Ax + \sum_{k=1}^{m} B_k u_k x + Cu \qquad (1)$$

where x is the n-dimensional state vector; u is the m-dimensional control vector; A, B_k (k=1,..., m), and C are matrices of appropriate dimensions which are usually assumed to be constant. Certain studies do consider variations of this model to include discrete systems, distributed systems, added constant vectors, state noise and control noise.

It is apparent that the addition of the product terms between state and control give an otherwise linear system a simple variable structure, adaptive control, or parametric control.

History

Early work on parametric control was reported on stability and switching systems in the Soviet Union and on circuit problems in the United States. Around 1962 this author, in conjunction with research on optimal control of nuclear reactors, investigated the significance of this class of systems which he termed bilinear. It was apparent that bilinear control processes are not only relevant to engineering systems but also to biology and economics. In some cases bilinear mathematical models arise in a natural manner from basic phenomena, and in others they represent another degree of approximation beyond that of linear models.

Details of much of this research is given by Mohler [1] and in U.S.-Italy Seminar Proceedings edited by Mohler and Ruberti [2,3].

MODELS AND IDENTIFICATION

Population Models

The simple biological population equation with birth rate minus death rate controlled is the simplest bilinear system. Feedback control such as evolve from competing species and predator - prey models, however, lead to more complex closed-loop nonlinear systems.

The neutron population, which is nearly proportional to power, in a controlled fission process may be approximated by

$$\frac{dn}{dt} = \frac{u(1-\beta)-1}{\ell} n + \sum_{i=1}^{6} \lambda_i c_i \tag{2}$$

$$\frac{dc_i}{dt} = \frac{u\beta_i}{\ell} n - \lambda_i c_i \;, \ldots, 6, \tag{3}$$

where u is neutron multiplication, ℓ is average neutron generation time, β_i is the portion of neutrons generated by precursors (unstable fission products) of population c_i $(i=1,\ldots,6)$, $\sum_{i=1}^{6} \beta_i = \beta$, and λ_i is the decay constant of the ith precursor [4]. It is interesting that biological cell populations behave in an analogous manner with some cells like neutrons born promptly, some born in a delayed reaction from precursors and still others dying or leaving the system. In fact, cell division, water drop division and nuclear fission are basically the same phenomenon with some necessary activation energy to overcome particle binding energy in one case and surface tension in the other. It should be noted here that both processes are naturally distributed, are stochastic diffusion processes, and that (2) and (3) only represent such crude approximations as are valid for overall control and performance studies.

Of much slower dynamics than nuclear fission, the buildup of poison products in a nuclear reactor is also a bilinear process as is approximately the conductive-convective heat-exchange process [4].

Biology

In the biological cell, bilinear control is synthesized by catalysts in chemical reactions which are presumed to obey the law of mass action. In this manner, the reaction rate is proportional to concentrations of reacting agents with a catalyst such as enzyme concentration entering as bilinear control. Numerous cascades of such enzyme controlled reactions take place in body metabolism [1,5]. For the transfer of material through the cell wall approximately bilinear control again is synthesized by enzyme concentration-in this case to vary membrane permeability.

Cells and large combinations of cells forming organs or convenient anatomical sections of organs may sometimes be investigated by compartmental models. For example, if a single substance or quantity X is distributed among n compartments, the rate of change of substance amount in the ith compartment is described by the conservation equations

$$\frac{dx_i}{dt} = \sum_{j=1}^{n}{}' \phi_{ij}(t) - \sum_{k=1}^{n}{}' \phi_{ki}(t) + \phi_{ia}(t) - \phi_{ai}(t) + p_i(t) - d_i(t) \tag{4}$$

where primed summation indicates deletion of ith term, ϕ_{ij} is flux of X from compartment j to compartment i, ϕ_{ia} (ϕ_{ai}) is the flux into (out of) compartment i from (to) the environment, p_i (d_i) is the rate of production (destruction) of χ in compartment i. It is easily shown that (4) may be reduced to the bilinear system (1) with an added n-vector. χ, for example, may be mass, heat or temperature.

Such compartmental models and the use of radioactive tracer experiments for system realization are presented by Mohler [1,6]. It is shown that specific tracer activity is related to the original compartmental state by means of an adjoint equation. Properties of the tracer dynamics are analyzed by Smith and Mohler [7]. Concepts of controllability and observability are found useful to determine necessary compartment accessibility (tracer insertion and observation), for minimum system realizations [1,2].

Realizations of a two-compartment water-balance model is developed from physiological concepts and tritiated water experiments on a wild house mouse in reference [6]. For water balance it is discovered that approximately bilinear control arises from the action of antidiuretic hormone (ADH) on tubule and collecting duct permeability in the kidney. Similar to temperature regulation, water balance is also regulated in an approximately bilinear manner by a network of constrictor muscles acting on renal plasma flow rate. Similarly water loss through skin and from lungs are manipulated parametrically by vasomotor control of blood flow, by skin permeability and by respiratory rate of lungs.

Bilinear models also are derived for control of respiration, for cardiac regulation and for temperature regulation [1,8].

A very interesting recent contribution on immune response models has been made by Bruni, Giovenco, Koch, and Strom [9]. They develop a nonlinear dynamical model of humoral immune response to the introduction of alien material (antigen) which is based on clonal selection theory. Association constants of cell receptors determine binding with a particular antigen according to some probability distribution. Briefly, the process involves different classes of cells with different purposes in the overall immune process. Immunocompetent cells are those with receptors for binding a particular antigen according to its association constant. Upon stimulation, these cells divide and some differentiate into plasma cells which further secrete antibodies and cannot divide further. Those which remain as immunocompetent cells are termed memory cells.

It is assumed that the free binding energy between a given antigen determinant and a cell receptor is Gaussian over some finite interval of association constants. This in turn leads to a log normal process for the association constant itself since changes in binding energy vary as the log of the association constant. The probability that a certain number of receptors out of the total per cell is described by Bernouli's distribution.

Again, the process has some analogy to the nuclear fission bilinear process. In this case some antibodies are born promptly, some have delayed birth through plasma cells or precursors and some are lost or die. Control takes place through the genetic code passed along successive generations and according to the particular antigen characteristic. From a control point of view the so-called association constant provides the mechanism of parametric control just as multiplication constant does for a nuclear fission process. While the fission process, (2) and (3), appear to be deterministic in nature, it should be realized that the equations in reality only represent averages - usually over an extremely large number of fissions. Basically, the probability of a neutron fissioning a nucleus is analogous to the probability of an antigen stimulating the immune response-both taking place through so-called association constants or multiplication constants.

It should be noted, however, that the above interpretation is that of this author and the blame for its crudeness should not be cast on the respected Italian researchers. The model derived in reference [9] includes nonlinear integral-differential equations which are even distributed with respect to association constant. The authors show, however, that their model may be interpreted as a bilinear system in a nonlinear nondynamic feedback loop.

Biondi [10] uses variable-structure models of the human acoustical system for diagnosis and prosthesis. The use of parametric control apparently has led to the development of an effective hearing device.

Socioeconomics

Parametric control may appear in economic processes as tax rates or as adjustable multipliers in numerous manners. Even socioeconomic compartments may arise in the form of administrative districts. Multiplicative control through migration rate may be manipulated by attractiveness multipliers such as job availability, housing and transportation.

A bilinear macroeconomic model is developed for the production of capital goods in a single sector by d'Allesandro [11]. With production a linear function of capital goods and labor, it is shown that capital goods x_1 and labor force x_2 are described by (1) with a zero C matrix,

$$A = \begin{bmatrix} -\mu & 0 \\ 0 & r \end{bmatrix} \quad \text{and } B = \begin{bmatrix} \alpha & \beta \\ 0 & 0 \end{bmatrix}$$

where u, scalar control, is the current saving ratio, μ is capital depreciation, and α, β are positive production multipliers. More complicated bilinear models of the same basic form are derived by disagregation of the previous example to geographical sectors or compartments. This compartmental model then includes the possibility of exchanging surplus investable resources, (as savings ratio above) from one region to another to synthesize bilinear controls. Similar models may also be derived by disaggregating with respect to different economic activities.

Aoki [12] develops a model for growth of capital stocks with tax rates on savings and consumption as a bilinear control.

System Realization and Identification

The construction of bilinear state space descriptions from input/output maps have been investigated by several researchers recently. Partial realization requires the matching of a finite sequence of constant parameters while complete realization requires matching over an infinite sequence which uniquely determines a given map. Isidori [13] derives explicit existence criteria and develops an algorithm for computing minimal realization in both cases. The complete realization study involves a generalization of linear system realization by the use of a suitable infinite Hankel matrix of input/output parameters. The partial realization involves a finite Hankel matrix which is a submatrix of that used for complete realization. Marchesini and Picci [14] develop some abstract theoretical aspects of multilinear system realization.

The general problem of parameter and state identification has not yet been solved. Balakrishnan [15] does show that linear system identification theory is sufficient for parameter identification of bilinear systems with additive state noise and additive output noise. A sequential maximum likelihood algorithm is developed which converges under certain conditions.

Willsky and Marcus [16] discuss the significance of bilinear estimation problems for phase-lock-loops and noisy oscillators in communication of an orbiting body and for orientation of a rotating rigid body. They use the theory of Lie groups and harmonic analysis to estimate the state of certain bilinear processes for which these methods are convenient.

SYSTEM PROPERTIES

Controllability

Except for some of the obvious properties of bilinear systems, controllability was the first basic property to be investigated in any detail. Numerous necessary conditions were derived and some of these are discussed by Mohler [1]. It is immediately obvious that simple results, such as the classic rank tests for controllability of linear systems, are not forthcoming for bilinear systems. Still, the adaptive nature of these systems makes their controllability of great practical use. Of course, this must be one reason for their prevalent role in nature [17].

It is shown in reference [17] by the existence of a Liapunov function that an asymptotically stable linear system with magnitude constrained control cannot be completely state controllable. It is readily seen, on the other hand, that such is not necessarily the case for bilinear systems. Sometimes it may be convenient to add a bilinear mode of control to an otherwise linear control system in order to improve its controllability, and, as will be discussed later, its performance as well. Mohler and Rink [18] utilize a local controllability result from Lee and Markus [19] along with the fact that controllability is really state connectedness to derive sufficient conditions for complete controllability of control constrained bilinear systems. The most convenient result applies to bilinear systems in the standard canonical or phase variable form. Here, the main result is that the phase-variable form of bilinear system (1) with $|u| \leq 1$ is completely state controllable if: 1) C is a nonzero matrix, and 2) all system eigenvalues can be shifted across the imaginary axis of the complex plane without passing through zero as u ranges continuously over the admissible set.

The equilibrium set and examples are discussed in reference [18], and properties of the reachable zone for equicontinuous control is analyzed by Mohler and Rink [20]. Besides showing the controllability assets of bilinear systems these early results demonstrate the inherent relation between the basic properties of controllability and stability.

Brockett [2]] and Elliott, Tarn and Goka [22,23] use Lie algebra to analyze a class of homogeneous bilinear systems. More recently Chang, Elliott and Tarn [24] present improved sufficient conditions for complete controllability of both discrete and continuous homogeneous bilinear systems with bounded control.

Kucera [25] was the first to analyze a particular class of control-constrained bilinear systems with the theory of Lie groups. Kucera shows that for his problem, the attainable states form a maximal integral manifold and that they can be reached with piecewise constant control of values -1,0, 1 for an admissable set defined such that $\sum_{k=1}^{m} |u_k| = 1$.

At a very early time, Caratheodory [26] introduced Pfaffian equations which Hermes [27] shows are quite naturally associated with systems which are linear in control. Controllability is derived from Pfaffian integrability by Hermes [27].

Structure

Brockett [19,28] presents the algebraic structure of bilinear systems and discusses their dynamical decomposition by means of Lie algebra. Then, these concepts are applied conveniently to controllability and observability analysis.

Bruni, DiPillo and Koch [29] study the structure of bilinear systems and their relation to higher-order multilinear systems. The Volterra series is studied for bilinear systems in this work. From this result along with the application of the reversion technique for the functional series expansion of the bilinear systems, it is seen that the Volterra kernels may all be synthesized by a sequence of linear operations which are characterized by the first-order kernel or impulse response.

Minimal realizations of the bilinear systems (1) from symmetrical kernels are studied by d'Allesandro, Isidori and Ruberti [30]. They show a convenient decomposition of the output solution as the sum of the zero-state response, the zero-input response and a coupled response represented by its Volterra series.

OPTIMAL CONTROL

Optimal bilinear regulation without the additive control term (i.e., zero C matrix in (1)), and the derivation of sufficient conditions are presented by Mohler and Rink [31]. In reference [32], these sufficiency conditions are applied to a problem with a non-unique maximum principle solution. While the method provides only optimal regulation, a comparison with linear system design is made for a servo tracking problem in reference [31]. The suboptimal bilinear servo shows far superior performance for the examples considered.

Since many practical control problems involve state space constraints, linear programming algorithms and their convenient software packages were investigated for bilinear control system design. A so-called quasilinear programming algorithm was developed which starts with an assumed optimal trajectory (e.g., maximum principle solution to reduced problem). The system equations and constraints are discretized and linearized about the assumed trajectory followed by the application of a standard linear programming algorithm. The process is repeated after the first iteration using the new optimal trajectory, and the problem is solved sequentially. Convergence cannot always be guaranteed, but such consideration is given to similar algorithms by Rosen [33]. The method is applied successfully to a complex nuclear reactor startup problem by Mohler, Moon, and Price [34].

An algorithm, which utilizes the gradient of the cost functional with respect to switching times, is developed by Mohler and Moon [35] in order to compute optimal bang-bang controls sequentially for linear-in-control systems. Obviously, the method is most amenable to bilinear systems since optimal bang-bang control is so prevalent for them and in which case the exponential state transition matrix is used. Convergence of this switching-time-variation method, STVM, to a maximum principle solution is shown [35]. The method is extended to fuel-optimal control computations in reference [36].

Singular controls are quite common for optimal bilinear control problems. This problem is given special consideration in the discrete case by Tarn [37].

Numerous applications to nuclear reactors are reported in reference [4]. These include time-optimal, coolant-optimal and minimal poison controls with magnitude and rate constraints on control and in some cases on state. Such practical considerations as terminal control synthesis, singular control synthesis, suboptimal closed-loop control and sliding control are developed [4].

Bilinear models appear in industrial chemical processes just as they do in the human body. Hofer [38] derives the optimal catalyst blend for tubular reactors and applies it to the conversion of hexadecane into octane.

ACKNOWLEDGEMENT

The author gratefully acknowledges the support of the National Science Foundation for this work. Research on bilinear systems in general have been boosted by means of the U.S.-Italy Cultural Program.

REFERENCES

1. Mohler, R.R., Bilinear Control Processes, Academic Press, New York, 1973.
2. Mohler, R.R., and Ruberti, A., Eds., Theory and Application of Variable-Structure Systems, Academic Press, New York, 1972.
3. Ruberti, A., and Mohler, R.R., Eds., Variable-Structure Systems with Application to Biology and Economics, Springer-Verlag, New York, forthcoming.
4. Mohler, R.R., and Shen, C.N., Optimal Control of Nuclear Reactors, Academic Press, New York, 1970.
5. Banks, H.T., "Nonlinear Time Varying Systems in Models for Enzyme Cascades," in Variable Structure Systems with Application to Biology and Economics (Ruberti and Mohler, Eds.), Springer-Verlag, New York, forthcoming.
6. Mohler, R.R., "Biological Modeling with Variable Compartmental Structure," IEEE Trans. Auto. Control, Vol. AC 19, No. 6, 1974 (forthcoming).
7. Smith, W.D., and Mohler, R.R., "New Techniques on the Tracer Analysis of Compartmented Biosystems," Proc. Third Annual Houston Conf. on Computer & System Science, 1971 (pp. 132-143).
8. Grodins, F.S., Control Theory and Biological Systems, Columbia Univ. Press, New York, 1963.
9. Bruni, C., Giovenco, M., Koch, G. and Strom, R., "Variable Structure Model of the Immune Response," in Variable Structure ...., (Ruberti and Mohler, Eds.), op. cit.
10. Biondi, E., "Application of Mathematical Models of Acoustical System Diagnosis and Prosthesis," ibid.
11. d'Allesandro, P., "Bilinearity and Sensitivity in Macroeconomy," ibid.
12. Aoki, M., "Some Examples of Dynamic Bilinear Models in Economics," ibid.
13. Isidori, A., "Direct Construction of Minimal Bilinear Realizations from Nonlinear Input-Output Maps," IEEE Trans. Auto. Control, Vol. AC18, 626-630, 1973.

14. Marchesini, G. and Picci, G., "Some Results on the Abstract Realization Theory of Multilinear Systems," in Theory and Application..., (Mohler and Ruberti, Eds.,), op.cit. pp. 109-136.

15. Balakrishnan, A.V., "Modeling and Identification Theory," ibid. pp.1-24.

16. Willsky, A.S. and Marcus, S.I., "Estimation for Bilinear Stochastic Systems," in Variable Structure..., (Ruberti and Mohler, Eds.), op.cit.

17. Mohler, R.R., "Natural Bilinear Control Processes," IEEE Trans. Sys. Sci. and Cyber., Vol. SSC6, 192-197, 1970.

18. Lee, E.B. and Markus,L., "Optimal Control for Nonlinear Processes," Arch. Rat. Mech. Anal., Vol. 8, 36-58, 1961.

19. Rink, R.E. and Mohler, R.R., "Completely Controllable Bilinear Systems," SIAM J. Control, Vol. 6, 477-486, 1968.

20. Mohler, R.R. and Rink, R.E., "Reachable Zones for Equicontinuous Bilinear Control Processes," Intl. J. Control, Vol. 14, 331-339, 1971.

21. Brockett, R.W., "System Theory on Group Manifolds and Coset Space," SIAM J. Control, Vol. 10, 265-284, 1972.

22. Elliott, D.L. and Tarn, T.J., "Controllability and Observability for Bilinear Systems," SIAM Natl. Meeting, Seattle, WA, 1971.

23. Tarn, T.J., Elliott, D.L. and Goka, T., "Controllability of Discrete Bilinear Systems with Bounded Control," IEEE Trans. Auto, Control, Vol. AC 18, 298-301, 1973.

24. Cheng, G.J., Tarn, T.J. and Elliott, D.L., "Controllability of Bilinear Systems," in Variable Structure..., (Ruberti and Mohler, Eds.),op.cit.

25. Kucera, J., "Solution in Large of Control Problem $\dot{x} = [A(1-u)+Bu]x$" , Czecholovak Math. J., Vol. 16, 600-622, 1966.

26. Caratheodory, C., "Calculus of Variations and Partial Differential Equations of First Order," Holden-Day, San Francisco, California, 1967. (Translated by J.L. Brandstatter of German Ed., published by Teubner, Berlin, 1935.)

27. Hermes, H., "Controllability and the Singular Problem," J. SIAM Control, Vol. 2, 241-260, 1965.

28. Brockett, R.W., "Algebraic Structure of Bilinear Models," in Theory and Application..., (Mohler and Ruberti, Eds.).

29. Bruni, C. DiPillo, G. and Koch, G., "Bilinear Systems: An Appealing Class Nearly Linear Systems in Theory and Application," IEEE Trans. Auto. Control, Vol. AC 19, 334-348, 1973.

30. d'Allesandro, P., Isidori, A. and Ruberti, A., "Realization and Structure Theory of Bilinear Dynamical Systems," in Theory and Application.., (Mohler and Ruberti, Eds.), op.cit. pp. 25-36.

31. Mohler, R.R. and Rink, R.E., "Control with a Multiplicative Mode," J. Basic Engr., Vol. 91, 201-206, 1969.

32. Mohler, R. R. and Rink, R.E., "Multivariable Bilinear System Control," Proc. IFAC Symp. Multivariable Control Sys., Dusseldorf, 1968.

33. Rosen, J.B., "Iterative Solution of Optimal Control Problems," SIAM J. Control, Vol. 4, 233-244, 1966.

34. Mohler, R.R., Moon, S.F. and Price, H.J., "Optimal Control Computations for Nuclear Reactors," in Computing Methods in Optimization Problems II (Zadeh, Neustadt and Balakrishnan, eds.), Academic Press, New York,1969.

35. Mohler, R.R. and Moon, S.F., "A Switching Time Variational Method for Nonlinear Optimal Control Problems," Ricerche di Automatica, Vol. 4, No. 2, 1973 (forthcoming).

36. Mohler, R.R. and Moon, S.F., "Fuel Optimal Control of a Class of Nonlinear Systems," Ricerche di Automatica, Vol. 4, No. 3, 1973 (forthcoming).

37. Tarn, T.J., "Singular Control of Bilinear Discrete Systems," Information and Control," Vol. 21, 211-234, 1972.

38. Hofer, E.P., "The Optimization of Bifunctional Catalysts in Tubular Reactors," Proc. Joint Auto. Control Conf., Columbus, 1973.

OPTIMAL CONTROL ON LIE GROUPS

JOHN BAILLIEUL
Division of Engineering and Applied Physics
Harvard University
Cambridge, Massachusetts

ABSTRACT

Classical optimization techniques are used to attack optimal control problems on Lie groups. Using methods arising from the calculus of variations, explicit solutions to certain optimization problems are found.

1. INTRODUCTION

It is the purpose of this paper to investigate the implications of classical optimization techniques as they apply to systems defined on Lie groups. In section 2 we discuss the Euler-Lagrange equation and the Pontryagin maximum principle in the context of Lie groups. (A more general discussion of the calculus of variations may be found in [5].) The results in this paper which are new include theorem 3 in section 3 which gives sufficient conditions for the existence of optimal controls and which yields a differential equation which must be satisfied by any optimal control. This generalizes earlier work of Brockett (see [2], p. 219). Also new in this paper are the results of section 4 in which we show that for some problems this differential equation can be solved in closed form, making it possible to explicitly compute the form of optimal controls. In section 5 we use the results of section 4 to attack a certain class of control problems which are not easily handled directly from the point of view of the calculus of variations. Finally, we work out the details of an example which illustrates some interesting behavior of optimal trajectories and the minimum return functions.

2. CLASSICAL OPTIMIZATION TECHNIQUES APPLIED TO LIE GROUPS

Suppose we have a system, defined on a Lie group G, whose evolution equation is given by

$$\frac{dx}{dt}(t) = \sum_{i=1}^{m} u_i(t)X_i(x(t)) \qquad t \in [0,T]; \quad x(0) = x_0 \qquad (1)$$

Here the u_i are piecewise continuous real valued functions, and the X_i are right invariant vector fields on G. In this section we consider the problem of finding control functions $u_1,\ldots,u_m$ which drive the state of the system from $x_0 \in G$ at time 0 to $x_1 \in G$ at time T in such a way as to minimize a cost functional

$$\eta = \int_0^T \tilde{L}(u_1(t),\ldots,u_m(t))dt ,$$

where $\tilde{L}$ is some suitably differentiable function on $\mathbb{R}^m$.

To facilitate the direct application of the calculus of variations we make the following assumptions about our problem. The first is that the function L may be expressed in terms of x(t) and dx(t)/dt. That is, we shall assume that for x(t), dx(t)/dt and u(t) related by (1), there is a

---

This research was supported by the U.S. Office of Naval Research under the Joint Services Electronics Program by Contract N00014-67-A00298-0006 and by a National Institutes of Health Fellowship 5 F01 HS 51,720-03.

function $L: T(G) \to \mathbb{R}$ (where $T(G)$ is the tangent bundle of G) such that $\tilde{L}(u_1,\dots,u_m) = L(x, dx/dt)$. The cost criterion we seek to minimize is then given by

$$\eta = \int_0^T L(x, \frac{dx}{dt})dt . \tag{2}$$

The second assumption is that there is an inner product $< , >_g$ defined for each $g \in G$ on the tangent space $T_g(G)$. $< , >_g$ is assumed to be smooth in the sense that if X,Y are smooth vector fields on G, then $<X_g,Y_g>_g$ is smooth as a function of g. The third assumption to be made is that for each $g \in G$ we have a smooth mapping ρ_g of $T_g(G)$ onto $T_e(G)$, where e is the group identity, and ρ_g serves to translate the inner product $< , >_g$ on $T_g(G)$ to $< , >_e$ on $T_e(G)$ as follows. Let $r_g : G \to G$ be the usual right translation mapping $h \mapsto hg$. Then we shall assume that for all $v, w \in T_g(G)$

$$<v,w>_g = <\rho_g(v),dr_{g-1}(w)>_e .$$

In the special case that $< , >_g$ is right invariant we simply set $\rho_g = dr_{g-1}$.

Next we assume without loss of generality that there is one local coordinate system $(U; x_1,\dots,x_n)$ on G such that for all $t \in [0,T]$ $x(t)$ remains in the coordinate neighborhood U. Thus our calculations will have the appearance of being carried out in $\mathbb{R}^n$. For brevity we shall write $x_i(t)$ for $x_i \circ x(t)$. Then in terms of local coordinates we have $L = L(x_1,\dots,x_n; \dot{x}_1,\dots,\dot{x}_n)$ where the $\dot{x}_i$'s are functions such that

$$\frac{dx(t)}{dt} = \sum_{i=1}^{n} \dot{x}_i(t) \left.\frac{\partial}{\partial x_i}\right|_{x(t)}$$

Finally let H be a Lie subgroup of G, which may be equal to G. Write $T_e(G) = T_e(H) + T_e(H)$ as an orthogonal direct sum decomposition with respect to the inner product $< , >_e$. Let $P : T_e(G) \to T_e(H)$ be the canonical projection onto the first factor of this decomposition. Then we have the following.

<u>Theorem 1</u>: With H and G as above, suppose $x(t) \in G$ is a path which remains entirely in H for $t \in [0,T]$ and $x(0) = x_0$ and $x(T) = x_1$. Then a necessary condition for $x(t)$ to locally minimize the functional η in (2) with respect to all paths joining x_0 and x_1 in H is that the Euler-Lagrange equation

$$0 = P\{\rho_{x(t)}[(\nabla_x L - \frac{d}{dt}\nabla_{\dot{x}}L)\Big|_{(x(t),\, \frac{dx(t)}{dt}}]\} \tag{3}$$

hold. Here $\nabla_x L$ and $\frac{d}{dt}\nabla_{\dot{x}}L$ are gradient vector fields on G which may be expressed in terms of our local coordinates as

$$\nabla_x L = \sum_{i=1}^{n} \frac{\partial L}{\partial x_i}\frac{\partial}{\partial x_i}$$

$$\frac{d}{dt}\nabla_{\dot{x}}L = \sum_{i=1}^{n} (\frac{d}{dt}\frac{\partial L}{\partial \dot{x}_i})\frac{\partial}{\partial x_i}$$

Theorem 1 is easily specialized to matrix Lie groups. Let $G=GL(n,\mathbb{R})$ and make the usual identifications of $\mathfrak{gl}(n,\mathbb{R})$ with tangent spaces to $GL(n,\mathbb{R})$. Let $< , >$ denote the usual matrix inner product $<A,B>=tr(A^tB)$.

Finally let $\rho_B : \mathfrak{gl}(n,\mathbb{R}) \to \mathfrak{gl}(n,\mathbb{R})$ be given by $\rho_B(A) = A^tB$ for $B \in GL(n,\mathbb{R})$. The equation (3) can be rewritten as

$$0 = P\{(\nabla_x L - \frac{d}{dt} \nabla_{\dot{x}} L)^t X(t)\} \quad .$$

It will be seen in the following sections that the Euler-Lagrange equation will allow us to explicitly compute optimal controls for certain problems. Unfortunately, such explicit results are limited to those optimal control problems amenable to treatment via the calculus of variations. An alternative approach to optimization is provided by the Pontryagin maximum, which we now present in terms of systems defined on matrix Lie groups.

Suppose G is a matrix Lie group with corresponding matrix Lie algebra $\mathfrak{g}$. Consider the system defined on G whose evolution equation is

$$\dot{X}(t) = (A + \sum_{i=1}^{r} u_i B_i)X(t); \quad A, B_1, \ldots, B_r \in \mathfrak{g} . \tag{4}$$

We wish to find a set of controls $u_1, \ldots, u_r$ which steers the system (4) from $X_o \in G$ to $X_1 \in G$ in T units of time in such a way as to optimize the cost functional

$$\eta = \int_0^T \sum_{i=1}^{r} \beta_i u_i^2 dt \quad , \tag{5}$$

where the β_i are positive real numbers.

Theorem 2: For the above optimization problem the Pontryagin maximum principle asserts that if u(t) is an optimal control, then there exists a matrix P such that

$$\dot{P}(t) = -({}^tA + \sum_{i=1}^{r} u_i(t) {}^tB_i)P(t)$$

and that the Hamiltonian H, defined by

$$H(P,X,u) = \langle P, (A + \Sigma u_i B_i)X\rangle + \lambda \sum_{i=1}^{r} \beta_i u_i^2 \quad , \tag{6}$$

where λ is some non-negative real, is minimized with respect to u by the optimal control.

Example 1: Consider a system on SO(n), the group of nxn orthogonal matrices, whose evolution equation is given by (4), where $A, B_1, \ldots, B_r$ are skew-symmetric. We wish to find that control u(t) which steers the state of this system from X_o at time 0 to X_1 at time T in such a way as to minimize the cost (5). According to the maximum principle we find that there is a matrix P satisfying

$$\dot{P}(t) = (A + \sum_{i=1}^{r} u_i(t)B_i)P(t) \quad ,$$

and the Hamiltonian for this problem is given by

$$H(X,P,u) = \langle P, (A + \Sigma u_i B_i)X\rangle + \sum_{i=1}^{r} \beta_i u_i^2 \quad .$$

(We have assumed here that λ appearing in (6) may be normalized to be 1.

This must be checked in individual cases.) Differentiating H with respect to u and setting the result equal to zero shows that the optimal control is given by $u_i = -\frac{1}{2\beta_i}\langle P^tX, B_i\rangle$. Since P and X satisfy the same linear differential equation and since X_o is non-singular, there is a constant matrix M such that $P(t) = X(t)M$. Hence we may rewrite our optimal control as

$$u_i(t) = -\frac{1}{2\beta_i}\langle X(t)M^tX(t), B_i\rangle .$$

3. OPTIMAL CONTROL ON MATRIX LIE GROUPS

Among the most important Lie subgroups of GL(n) to which the method of theorem 1 may be applied are the symplectic group SP(2m), where 2m=n, and the compact real forms of the classical linear groups: SO(n), SU(n) and $Sp(2m) \cap U(2m)$. To expedite our computations for SU(n) and $Sp(2m) \cap U(2m)$ we shall work with their realifications. The realification of any Lie subalgebra of $\mathfrak{gl}(n,\mathbb{C})$ is obtained by suitable restriction of the Lie algebra homomorphism $\mathfrak{gl}(n,\mathbb{C}) \to \mathfrak{gl}(2n,\mathbb{R})$ which is given by

$$B \longrightarrow \begin{bmatrix} \mathrm{Re}\,B & \mathrm{Im}\,B \\ -\mathrm{Im}\,B & \mathrm{Re}\,B \end{bmatrix}$$

We denote the realification of su(n) by $su(n)_{\mathbb{R}}$ (resp. $sp(n) \cap u(n)$ by $[sp(n)]_{\mathbb{R}}$), and the corresponding group realification $SU(n)_{\mathbb{R}}$ (resp. $Sp(n)_{\mathbb{R}}$) is obtained by exponentiating $su(n)_{\mathbb{R}}$ (resp. $sp(n)_{\mathbb{R}}$).

Assume in what follows that we have a fixed subgroup $H \leq GL(n,\mathbb{R})$, and let $\mathfrak{h}$ be the corresponding subalgebra of $\mathfrak{gl}(n,\mathbb{R})$. Let us consider systems defined on H whose evolution equations are given by

$$\dot{X}(t) = (A + \sum_{i=1}^{r} u_i(t)B_i)X(t) \tag{7}$$

where $A, B_1,\ldots,B_r \in \mathfrak{h}$, and for which we have a performance criterion of the form

$$\eta = \int_0^T \mathrm{tr}[K(\Sigma u_iB_i)K(\Sigma u_i{}^tB_i)]dt \quad , \tag{8}$$

where K is an nxn constant, symmetric positive definite matrix of weighting coefficients. For brevity, write $\Omega(t) = \sum_{i=1}^{r} u_i(t)B_i$. Then we have the following theorem.

Theorem 3: If the linear span of $B_1,\ldots,B_r$ is $\mathfrak{h}$ then

(i) the system (7) is controllable in the sense that given any elements $X_0, X_1 \in H$, there exists a time T > 0 and a control matrix $\Omega(t) = \Sigma u_i(t)B_i$ which steers the system from X_0 at time 0 to X_1 at time T, and moreover

(ii) among all control matrices which accomplish this, there is at least one control $\Omega^o(t)$ for which (8) is minimized, and finally

(iii) if $A, B_1,\ldots,B_r$ are self-contragradient then $\Omega^o(t)$ satisfies the Euler-Lagrange equation

$$P_{\mathfrak{h}}(K\dot{\Omega}K) = P_{\mathfrak{h}}(K\Omega K^t(A+\Omega) - {}^t(A+\Omega)K\Omega K)$$

where $P_{\mathfrak{h}} : \mathfrak{gl}(n,\mathbb{R}) \to \mathfrak{h}$ is the orthogonal projection onto the first factor of $\mathfrak{gl}(n,\mathbb{R}) = \mathfrak{h} \oplus \mathfrak{h}^{\perp}$.

Proof: That the system is controllable follows from the results of Brockett, ([1]). That there exists an optimal control follows from [3], and the form of the Euler-Lagrange equation is a calculation based on theorem 1.

Corollary: If K is a diagonal matrix in the case that $\mathfrak{h} = so(n)$ or if K is a diagonal matrix of the form

$$K = \begin{bmatrix} K_1 & 0 \\ 0 & K_1 \end{bmatrix}$$

in the case that $\mathfrak{h} = su(m)_{\mathbb{R}}$ (with 2m=n) or if K is a diagonal matrix of the form

$$K = \begin{bmatrix} K_2 & & & \\ \cdot & K_2 & & \\ & & K_2 & \\ & & & K_2 \end{bmatrix}$$

in the case that $\mathfrak{h} = sp(m)$ (with 2m=n and m even) then the Euler-Lagrange equation in part (iii) of theorem 3 takes the form

$$\dot{\Omega} = K^{-1}(A+\Omega)K\Omega - \Omega K(A+\Omega)K^{-1} \qquad (9)$$

Remark: Cost functionals of the form (8) allow us to consider a fairly broad range of optimization problems. For example, suppose $\mathfrak{h}$ is the compact real form of one of the classical Lie algebras mentioned in the preceeding corollary. Let $\Omega(t)$ be one of the following: (i) an nxn skew-symmetric matrix if $\mathfrak{h} = so(n)$; (ii) a skew-symmetric matrix of the form

$$\begin{bmatrix} \Omega_{11} & \Omega_{12} \\ -\Omega_{12} & \Omega_{11} \end{bmatrix}$$

with Ω_{12} mxm, symmetric and traceless if $\mathfrak{h} = su(m)_{\mathbb{R}}$ (2m=n); (iii) a skew-symmetric matrix of the form

$$\left\{\begin{matrix} \Omega_{11} & \Omega_{12} & \Theta_{11} & \Theta_{12} \\ -\Omega_{12} & -{}^{t}\Omega_{11} & \Theta_{12} & -\Theta_{11} \\ -\Theta_{11} & -\Theta_{12} & \Omega_{11} & \Omega_{12} \\ -\Theta_{12} & \Theta_{11} & -\Omega_{12} & -{}^{t}\Omega_{11} \end{matrix}\right\}$$

with each entry an $\frac{m}{2} \times \frac{m}{2}$ matrix if $\mathfrak{h} = sp(m)_{\mathbb{R}}$ (2m=n; m even). Then for the system

$$\dot{X}(t) = \Omega(t)X(t) \qquad (10)$$

we may consider an arbitrary weighted quadratic performance measure

$$\eta = \int_0^T \sum_{i<j} b_{ij} w_{ij}(t)^2 dt . \qquad (11)$$

Define the symmetric matrix B to have ijth element b_{ij} (i<j) and let b_{ii} be arbitrary. Let A be a matrix square root of B (which always exists since B is symmetric). We may now rewrite (11) as

$$\eta = \int_0^T \operatorname{tr}[K(\Omega \otimes I + I \otimes \Omega)]^2 dt$$

where K is an $n^2 x n^2$ diagonal matrix whose entries are defined by

$$k_{i+(j-1)n} = \frac{1}{2} a_{ij} \qquad i,j=1,\ldots,n \quad .$$

Hence we simply consider $\mathfrak{h}$ as a subalgebra of $SO(n^2)$, and all our previous results allow us to compute an Euler-Lagrange equation for (11), except that here the calculations become somewhat messy. An example of the type of results to be obtained, however, may be found in taking $\mathfrak{h} = so(n)$. The resulting Euler-Lagrange equation may be written directly in terms of nxn skew-symmetric matrices as

$$\dot{\hat{\Omega}} = [\hat{\Omega},\Omega]$$

where the ijth entry in $\hat{\Omega}$ is $b_{ij}\omega_{ij}$. The resulting Euler-Lagrange equations for $su(m)_{\mathbb{R}}$ and $sp(m)_{\mathbb{R}}$ have the same form, but $\hat{\Omega}$ is more complicated.

4. SOLUTION OF THE EULER-LAGRANGE EQUATION

For the most general diagonal matrices K in each of the three cases mentioned in the corollary to theorem 3 and for $A \neq 0$, equation (9) does not admit a solution in terms of elementary functions. If we assume $A = 0$ and further restrict K in each case, however, it becomes possible to write down explicit solutions. For example, in case $\mathfrak{h} = so(n)$ assume that K takes the form

$$K = \left[\begin{array}{c|c} k\left\{\overbrace{\begin{matrix}1 & & \\ & \ddots & \\ & & 1\end{matrix}}^{k}\right. & \\ \hline & \begin{matrix}\alpha & & \\ & \ddots & \\ & & \alpha\end{matrix}\end{array}\right]$$

Then it is easy to see that $\Omega(t)$ is conformably partitioned into blocks

$$\begin{bmatrix} \Omega_{11} & \Omega_{12} \\ -{}^t\Omega_{12} & \Omega_{22} \end{bmatrix}$$

where Ω_{11} is a kxk skew-symmetric constant matrix and Ω_{22} is a (n-k)x(n-k) skew-symmetric constant matrix. $\Omega_{12}(t)$ is a kx(n-k) matrix which satisfies the differential equation

$$\dot{\Omega}_{12}(t) = (1-\frac{1}{\alpha})\Omega_{11}\Omega_{12}(t) + (\alpha-1)\Omega_{12}(t)\Omega_{22}$$

so that

$$\Omega_{12}(t) = e^{C_1 t} A_o e^{C_2 t}$$

where

$$C_1 = (1 - \frac{1}{\alpha})\Omega_{11}$$

$$C_2 = (\alpha-1)\Omega_{22}$$

$$A_o = \Omega_{12}(0)$$

Putting the results together, we find

$$\Omega(t) = e^{\begin{bmatrix} C_1 & \\ & {}^tC_2 \end{bmatrix} t} \Omega_o e^{\begin{bmatrix} {}^tC_1 & \\ & C_2 \end{bmatrix} t}$$

where

$$\Omega_o = \begin{bmatrix} \Omega_{11} & \Omega_{12}(0) \\ -{}^t\Omega_{12}(0) & \Omega_{22} \end{bmatrix}$$

If we write $C = \begin{bmatrix} C_1 & \\ & {}^tC_2 \end{bmatrix}$, then the above computations lead to the following.

Theorem 4: Consider the optimal control problem whose dynamics are given by (10) and for which the cost functional is η in (8) and K is as above. The optimal control matrix for this problem has the form

$$\Omega(t) = e^{Ct}\Omega_o e^{{}^tCt}$$

where C, Ω_o are constant matrices in so(n). Moreover, the optimal trajectory is expressable as

$$X(t) = e^{Ct}e^{(-C+\Omega_o)t}X_o$$

Corollary 1: For the optimal control problem considered in theorem 4, the optimal cost is

$$\eta = tr(K\Omega_o)^2 \cdot T$$

Corollary 2: Using the notation of the corollary to theorem 3, if K_1 is of the form

$$\left[\begin{array}{ccc|ccc} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ \hline & & & \alpha & & \\ & & & & \ddots & \\ & & & & & \alpha \end{array}\right]$$

(resp K_2 is of this form) then there are constant matrices $\Omega_o C \ \varepsilon \ su(m)_{\mathbb{R}}$ (resp. Ω_o, $C \ \varepsilon \ sp(m)_{\mathbb{R}}$) such that the optimal control matrix is given by

$$\Omega(t) = e^{Ct}\Omega_o e^{{}^tCt}$$

and the corresponding optimal trajectory on $SU(m)_{\mathbb{R}}$ (resp. $Sp(m)_{\mathbb{R}}$) is expressable as

$$X(t) = e^{Ct}e^{(-C+\Omega_o)t}X_o \quad .$$

5. RESTRICTION OF THE CONTROLS TO A SUBSPACE OF THE LIE ALGEBRA

In the optimization problems in the preceding sections we have placed no restrictions on the values the control matrix $\Omega(t)$ can take on in the Lie algebra $\mathfrak{h}$. We now consider a class of problems in which $\Omega(t)$ is restricted to take the values in a subspace $\mathfrak{k} \subseteq \mathfrak{h}$. We shall state results for the compact real forms of the classical linear groups, and for these will be defined as follows. If $\mathfrak{h} = so(n)$, $\mathfrak{k}$ will be the subspace of nxn matrices with nonzero entries only in the n-th row and n-th column. If $\mathfrak{h} = su(n)_{\mathbb{R}}$, $\mathfrak{k}$ will be the subspace of 2mx2n matrices with nonzero entries only in the n-th row and column and the 2n-th row and column. Finally, if $\mathfrak{h} = sp(n)_{\mathbb{R}}$ (with n even) $\mathfrak{k}$ will be the subspace of 2nx2n matrices with nonzero entries only in the n/2-th, n-th, 3n/2-th and 2n-th columns and rows. In each of these cases let $\mathfrak{k}^{\perp}$ denote the orthogonal complement of $\mathfrak{k}$ with respect to our inner product $< , >$. Then we note that any $\Omega \in \mathfrak{h}$ may be uniquely written as $\Omega = \Omega_a + \Omega_r$ with $\Omega_a \in \mathfrak{k}$, $\Omega_r \in \mathfrak{k}^{\perp}$. In the ensuing discussion, a matrix in $\mathfrak{h}$ which is also in $\mathfrak{k}$ will be called an admissible matrix.

Lemma: Consider the system whose evolution equation is given by

$$\dot{X}(t) = \Omega(t)X(t) \tag{13}$$

on the Lie group $H = SO(n)$, $SU(n)_{\mathbb{R}}$ or $Sp(n)_{\mathbb{R}}$. If $\Omega(t)$ is restricted to lie in $\mathfrak{k}$ for all $t \geq 0$, (this is to say $\Omega(t)$ is restricted to be an admissible control), then (13) is still controllable on H in the sense of theorem 3(i).

Proof: Let E_{ij} $(1 \leq i < j \leq n)$ denote the skew-symmetric matrix whose i,j-th entry is 1, whose j,i-th entry is -1 and all other entries of which are 0. Then the matrices E_{1j}, j=2,...,n, span the space of admissable matrices for SO(n). Since $E_{ij} = [E_{1i}, E_{1j}]$ for $i < j$, the matrices E_{1j} also generate the Lie algebra so(n). It may similarly be shown that by taking linear combinations of Lie brackets of admissible matrices for $SU(m)_{\mathbb{R}}$ (resp. $Sp(m)_{\mathbb{R}}$) we can generate the Lie algebras $su(m)_{\mathbb{R}}$ (resp. $sp(m)_{\mathbb{R}}$). Now controllability of (13) follows from [1], theorem 5.

The primary problem: For X_0, $X_1 \in H$, we wish to find an admissable control $\Omega(t)$ which drives the state of the system (13) from X_0 at time 0 to X_1 at time T so as to minimize the cost

$$\eta = -\frac{1}{2}\int_0^T tr\Omega(t)^2 dt \quad .$$

If $\Omega(t)$ is such an optimal admissible control, let η^{o} denote the corresponding optimal cost.

The auxiliary problems: For each real $\lambda > 0$ and X_0, X_1 given above we wish to find a control $\Omega_\lambda(t)$ in $\mathfrak{h}$ which drives the state of the system (13) from X_0 at time 0 to X_1 at time T so as to minimize the cost

$$\eta_\lambda = -\frac{1}{2}\int_0^T tr(\Omega_{\lambda a}^{2}(t)) + \lambda tr(\Omega_{\lambda r}^{2}(t))dt$$

(where we are writing $\Omega_\lambda = \Omega_{\lambda a} + \Omega_{\lambda r}$ in terms of the orthogonal decomposition $\mathfrak{h} = \mathfrak{k} + \mathfrak{k}^{\perp}$.) Let η_λ^{o} denote the corresponding optimal cost.

The auxiliary problems constitute a one parameter family of problems (parameterized by λ ranging over the positive reals) which it will turn out can be solved by the methods of the last section. The goal of this section is to show that in certain cases we may obtain a solution to the primary problem as a limit of solutions to the auxiliary problems.

To solve the λ-auxiliary problem we use the notation of the corollary to theorem 3 to rewrite η_λ as

$$\eta_\lambda = -\frac{\lambda}{2}\int_0^T \operatorname{tr}(K_\lambda \Omega_\lambda(t))^2 dt ,$$

where K_λ is a diagonal matrix of the form (12) with $k = n-1$ and $\alpha = 1/\lambda$ if $\mathfrak{h} = so(n)$; K_1 is of this form with $k = n-1$ if $\mathfrak{h} = su(n)_{\mathbb{R}}$; and K_2 is of this form with $k = \frac{n}{2} - 1$ if $\mathfrak{h} = sp(n)_{\mathbb{R}}$. The results of the preceding section show that the optimal $\Omega_\lambda(t)$ for this problem is of the form

$$\Omega_\lambda(t) = e^{(1-\lambda)B_\lambda t} A_\lambda e^{-(1-\lambda)B_\lambda t}$$

where $\Omega_\lambda(0) = A_\lambda + B_\lambda$ with $A_\lambda \varepsilon \mathfrak{k}$ and $B_\lambda \varepsilon \mathfrak{k}^\perp$. The corresponding optimal trajectory for this auxiliary problem is of the form

$$X_\lambda(t) = e^{(1-\lambda)B_\lambda t} e^{(\lambda\cdot B_\lambda + A_\lambda)t}$$

Theorem 5: Let H be one of SO(n), $SU(n)_{\mathbb{R}}$ or $Sp(n)_{\mathbb{R}}$ with $\mathfrak{h}$ the corresponding matrix Lie algebra. There exist an optimal admissible control $\Omega(t)$ which solves the primary problem. Moreover, if η^o is the corresponding optimal cost, and if $\lambda\cdot B_\lambda$ appearing in the above solution to the auxiliary problem tends to a limit B as $\lambda \to \infty$, then we may choose a sequence of positive real numbers $\{\lambda_i: \lambda_i \to \infty \text{ as } i \to \infty\}$ such that

(i) $\lim_{i\to\infty} \eta^o_{\lambda i} = \eta^o$

(ii) $\lim_{i\to\infty} A_{\lambda i} = A$ exists in $\mathfrak{k}$

(iii) $\eta^o = -\frac{1}{2}\cdot T\cdot \operatorname{tr}(A^2)$

(iv) $\lim_{i\to\infty} \Omega_{\lambda_i}(t) = e^{-Bt}Ae^{Bt}$ is an optimal admissible control which solves the primary problem

(v) $\lim_{i\to\infty} X_{\lambda_i}(t) = e^{-Bt}e^{(B+A)t}$ is the corresponding optimal trajectory.

The proof of this theorem is somewhat lengthy and will be omitted. The following example shows how these results may be applied to SO(3).

Example 2: Consider system (13) on SO(3) where $\Omega(t)$ is restricted to lie in $\mathfrak{k}$. This means $\Omega(t)$ is of the form

$$\begin{bmatrix} 0 & 0 & -\omega_2(t) \\ 0 & 0 & \omega_1(t) \\ \omega_2(t) & -\omega_1(t) & 0 \end{bmatrix}$$

We now wish to find the control $\Omega(t)\ \varepsilon$ of this form which drives the system from the identity at time t=0 to the matrix

$$X_1 = \begin{bmatrix} \sqrt{1-\varepsilon^2} & -\varepsilon & 0 \\ \varepsilon & \sqrt{1-\varepsilon^2} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

at time t=1, so as to minimize the performance measure

$$\eta = \int_0^T \omega_1(t)^2 + \omega_2(t)^2 dt = -\frac{1}{2}\int_0^T \mathrm{tr}\,\Omega(t)^2 dt \ .$$

ε is taken to be a small positive real number.

In light of the preceding theorem we wish to find real parameters $\omega, \omega_1, \omega_2$ such that we can write

$$X_1 = e^{\begin{bmatrix} 0 & \omega & 0 \\ -\omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}} e^{\begin{bmatrix} 0 & -\omega & -\omega_2 \\ \omega & 0 & \omega_1 \\ \omega_2 & -\omega_1 & 0 \end{bmatrix}}$$

Then the optimal control will be given by

$$\Omega(t) = \begin{bmatrix} 0 & 0 & -\lambda\cos(\omega t+\phi) \\ 0 & 0 & \lambda\sin(\omega t+\phi) \\ \lambda\cos(\omega t+\phi) & -\lambda\sin(\omega t+\phi) & 0 \end{bmatrix}$$

where $\lambda = \sqrt{\omega_1^2+\omega_2^2}$ and $|\phi|$ is the smallest number such that $\sin\phi = \omega_1/\lambda$. Hence we now compute values for ω, ω_1, ω_2 and η^o.

Let $c = \sqrt{\lambda^2+\omega^2}$ and $\omega_1' = \omega_1/c$, $\omega_2' = \omega_2/c$ and $\omega' = \omega/c$. Then it is easy to see

$$e^{\begin{bmatrix} 0 & -\omega & -\omega_2 \\ \omega & 0 & \omega_1 \\ \omega_2 & -\omega_1 & 0 \end{bmatrix}}$$

$$= \begin{bmatrix} \omega_1'^2+(\omega_2'^2+\omega'^2)\cos c & -\omega'\sin c+\omega_1'\omega_2'(1-\cos c) & -\omega_2'\sin c+\omega_1'\omega'(1-\cos c) \\ \omega'\sin c+\omega_1'\omega_2'(1-\cos c) & \omega_2'^2+(\omega_1'^2+\omega'^2)\cos c & \omega_1'\sin c+\omega_2'\omega'(1-\cos c) \\ \omega_2'\sin c+\omega_1'\omega'(1-\cos c) & -\omega_1'\sin c+\omega_2'\omega'(1-\cos c) & \omega'^2+(\omega_1'^2+\omega_2'^2)\cos c \end{bmatrix}$$

For our problem we must have

$$e^{\begin{bmatrix} 0 & -\omega & -\omega_2 \\ \omega & 0 & \omega_1 \\ \omega_2 & -\omega_1 & 0 \end{bmatrix}} = \begin{bmatrix} \cos\omega & -\sin\omega & 0 \\ \sin\omega & \cos\omega & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{1-\varepsilon^2} & -\varepsilon & 0 \\ \varepsilon & \sqrt{1-\varepsilon^2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (14)$$

Carrying out the multiplication and comparing matrix entries, we find $\omega_1'\sin c = \omega_2'\sin c = 0$. It is clearly impossible that $\omega_1' = \omega_2' = 0$, so we must have $c = k\cdot\pi$ for some integer k. Again comparing entries in the previous matrix equation, we find k is even so that $c = 2m\pi$ for some integer m. Hence (14) reduces to

$$\begin{bmatrix} \cos\omega & -\sin\omega & 0 \\ \sin\omega & \cos\omega & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{1-\varepsilon^2} & -\varepsilon & 0 \\ \varepsilon & \sqrt{1-\varepsilon^2} & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

so that $\cos\omega = \sqrt{1-\varepsilon^2}$, $\sin\omega = -\varepsilon$ and ω_1,ω_2 are arbitrary, subject to the proviso that $\omega_1^2+\omega_2^2+\omega^2 = 4m^2\pi^2$. Hence our optimization problem reduces to choosing m and ω such that $\sin\omega = -\varepsilon$ and such that $4m^2\pi^2-\omega^2$ (which is of course equal to the total cost η) is minimized.

Suppose we have succeeded in choosing m,ω such that this minimization occurs. Then we claim that

$$0 \leqslant 4m^2\pi^2-\omega^2 \leqslant 4\pi^2 ,$$

since if $4m^2\pi^2-\omega^2 > 4\pi^2$ we could obviously improve the cost by choosing $m=1$ and $0 \leqslant |\omega| \leqslant 2\pi$. Having obtained this bound on the cost $\eta = 4m^2\pi^2-\omega^2$, it is a simple but length argument to show that indeed the optimal cost is obtained by selecting $m=1$ and $-2\pi \leqslant \omega \leqslant 2\pi$.

REFERENCES

[1] R.W. Brockett, "System Theory on Group Manifolds and Coset Spaces," SIAM J. Control, 10 (1972), pp. 265-84.

[2] R.W. Brockett, "Lie Theory and Control Systems Defined on Spheres," SIAM J. Appl. Math., 25 (1973), pp. 213-25.

[3] L. Cesari, "Existence Theorems for Optimal Solutions in Lagrange and Pontryagin Problems," SIAM J. Control, 3, (1965), pp. 475-98.

[4] H. Samelson, Notes on Lie Algebras, Van Nostrand Reinhold, N.Y., 1969.

[5] S. Sternberg, Lectures on Differential Geometry, Prentice Hall, Englewood Cliffs, N.J., 1964.

Acknowledgement: The author wishes to express his thanks to Prof. R.W. Brockett for his guidance and encouragement during this research.

LINEARIZATION AND BILINEARIZATION OF CONTROL SYSTEMS

ARTHUR J. KRENER
Department of Mathematics,
University of California, Davis

ABSTRACT

The standard autonomous and time dependent control systems are first order approximations to what are generally nonlinear phenomena in neighborhoods of a point and reference trajectory respectively. This helps to explain mathematically the wide applicability of these models. In this paper we show that higher order approximations around a point and a reference trajectory naturally result in autonomous and time dependent bilinear control systems respectively. This augurs well for the theoretical and applied usefulness of these models which are now being extensively studied.

1. LINEARIZATION AROUND A POINT

The time evolution of many engineering processes can be effectively modelled by an autonomous control system of nonlinear type (1.1),

$$\begin{aligned} \dot{x} &= f(x,u), \\ y &= g(x,u), \\ x(0) = x^o \quad &\text{and} \quad |u_i - u_i^o| \leq c_i \ . \end{aligned} \tag{1.1}$$

Here the n vector x represents the state of the system, the h vector u the input or control, the k vector y the output, and x^o and u^o are initial values of the state and control.

In general the functions f and g, which are assumed to be as smooth as needed, may be extremely complex in their global dependence on x and u but the actual system may operate in a relatively narrow range of x and u values around x^o and u^o. For convenience we change x and u coordinates so that $x^o = 0$, $u^o = 0$ and $c_i = 1$ then (1.1) can be approximated by

$$\begin{aligned} \dot{x} &= f(0,0) + \frac{\partial f}{\partial x}(0,0)x + \frac{\partial f}{\partial u}(0,0)u \ , \\ y &= g(0,0) + \frac{\partial g}{\partial x}(0,0)x + \frac{\partial g}{\partial u}(0,0)u \ , \\ x(0) &= 0 \quad \text{and} \quad |u| \leq 1 \end{aligned} \tag{1.2}$$

For any admissible input u(t) let x(t), y(t) and $\overline{x}(t)$, $\overline{y}(t)$ denote the solution of (1.1) and (1.2) respectively. In general for small t we expect $|x(t)|$ and $|\overline{x}(t)|$ to grow like t and therefore, ignoring errors due to the truncation of terms in u, we expect that the right sides of (1.1) and (1.2) will differ by a term of order t^2 from which it follows that $|x(t)-\overline{x}(t)|$ will grow like t^3 and $|y(t)-\overline{y}(t)|$ like t^2.

If we introduce a new coordinate x_o which is identically one then (1.2) can be transformed into a linear system

$$\begin{aligned} \dot{x} &= Ax + Bu \ , \\ y &= Cx + Du \ , \\ x(0) &= (1,0,\dots,0), \quad |u| \leq 1 \ , \end{aligned} \tag{1.3}$$

where

$$A = \left[\begin{array}{c|c} 0 & 0 \quad\cdots\quad 0 \\ \hline f(0,0) & \frac{\partial f}{\partial x}(0,0) \end{array}\right] \qquad B = \left[\begin{array}{c} 0 \quad\cdots\quad 0 \\ \frac{\partial f}{\partial u}(0,0) \end{array}\right]$$

$$C = \left[\begin{array}{c|c} g(0,0) & \frac{\partial g}{\partial x}(0,0) \end{array}\right] \qquad D = \left[\begin{array}{c} \frac{\partial g}{\partial u}(0,0) \end{array}\right]$$

We have an impressive body of theory regarding such systems and their practical usefulness has been proven in a wide variety of applications. This seems to contradict our rough error analysis made above since it led us to expect errors that grow rather fast, like t^3. However that argument was predicated on the assumption that $|x(t)|$ is growing like t. In many real situations the processes being modeled are inherently stable so that (1.2) is asymptotically stable and any errors caused by the approximation decay rather than grow.

2. BILINEARIZATION AROUND A POINT

In order to generate approximations to (1.1) of higher accuracy we expand the right side in a series containing higher terms. We restrict to systems where the control u enters linearly and is a scalar. The generalization to vector controls is straightforward but if the dependence on u is nonlinear other techniques must be employed. With these restrictions (1.1) becomes

$$\begin{aligned} \dot{x} &= f(x,u) = f_o(x) + uf_1(x) \quad , \\ v &= g(x,u) = g_o(x) + ug_1(x) \quad , \qquad (2.1) \\ x(0) &= 0 \quad \text{and} \quad |u| \leqslant 1 \quad . \end{aligned}$$

If we truncate after the second order terms we obtain

$$\begin{aligned} \dot{x} &= f_o(0) + \frac{\partial f_o(0)}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 f_o(0)}{\partial x^2} x + uf_1(0) + u \frac{\partial f_1(0)}{\partial x} x + ux^T \frac{\partial^2 f_1(0)}{\partial x^2} x \\ y &= g_o(0) + \frac{\partial g_o(0)}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 g_o(0)}{\partial x^2} x + ug_1(0) + u \frac{\partial g_1(0)}{\partial x} x + ux^T \frac{\partial g_1(0)}{\partial x} x \qquad (2.2) \\ x(0) &= 0 \quad \text{and} \quad |u| \leqslant 1 \end{aligned}$$

These equations cannot be linearized, however by introducing several new state variables in a technique due to Carleman ([1] and [2]) we can put (2.2) into bilinear form

$$\begin{aligned} \dot{x} &= Ax + uBx \quad , \\ y &= Cx + uDx \quad , \\ x(0) &= (1,0,\ldots,0) \quad , \quad |u| \leqslant 1 \quad . \end{aligned}$$

To show this we proceed as follows, let $x^{[p]}$ denote the vectors whose components consists of all linearly independent monomials of degree p in the variables $x_1,\dots,x_n$, ordered lexographically. (A similar approach is found in Brockett, [3] and [4].) For example

$$x^{[2]} = (x_1^2, x_1x_2, \dots, x_1x_n, x_2^2, x_2x_3, \dots, x_n^2) \quad \text{and}$$

$$x^{[3]} = (x_1^3, x_1^2x_2, \dots, x_1^2x_n, x_1x_2^2, \dots, x_n^3) \quad .$$

Let $x^{[0]}$ be identically the scalar 1. Then $\dot{x}^{[0]} = 0$ and from (2.2)

$$\dot{x}^{[1]} = A_0^1x^{[0]}+A_1^1x^{[1]}+A_2^1x^{[2]}+uB_0^1x^{[0]}+uB_1^1x^{[1]}+uB_2^1x^{[2]} \tag{2.4}$$

where A_i^1 and B_i^1 are matrices of the appropriate dimensions easily computable from $\dfrac{\partial^i f_o(0)}{\partial x^i}$ and $\dfrac{\partial^i f_1(0)}{\partial x^i}$ respectively.

To develop a differential equation for $x^{[2]}$ consider a particular component x_ix_j. By the product rule, $d/dt(x_ix_j) = \dot{x}_ix_j+x_i\dot{x}_j$ and since $\dot{x}_i$ and $\dot{x}_j$ depend bilinearly on $x^{[0]}$, $x^{[1]}$, $x^{[2]}$ and u it follows that $d/dt(x_ix_j)$ depends bilinearly on $x^{[1]}$, $x^{[2]}$, $x^{[3]}$ and u,

$$\dot{x}^{[2]} = A_1^2x^{[1]}+A_2^2x^{[2]}+A_3^2x^{[3]}+uB_1^2x^{[1]}+uB_2^2x^{[2]}+uB_3^2x^{[3]} \quad . \tag{2.5}$$

The matrices A_i^2 and B_i^2 are computable from A_i^1 and B_i^1. In general $x^{[p]}$ satisfies a bilinear differential equation

$$\dot{x}^{[p]} = A_{p-1}^px^{[p-1]}+A_p^px^{[p]}+A_{p+1}^px^{[p+1]}+uB_{p-1}^px^{[p-1]}+uB_p^px^{[p]}+uB_{p+1}^px^{[p+1]} . \tag{2.6}$$

Truncating at terms of order 2 in the transition from (2.1) to (2.2) is equivalent to setting $x^{[p]} = 0$ for $p \geqslant 3$ so this is assumed in (2.5) and (2.6).

In a similar fashion the output map is expanded

$$y = C_ox^{[0]}+C_1x^{[1]}+C_2x^{[2]}+uB_1x^{[1]}+uB_2x^{[2]}+uB_3x^{[3]} \quad , \tag{2.7}$$

and the result is (2.3), a finite dimensional bilinear system.

In general the nonlinear system (2.1) gives rise to an infinite dimensional bilinear system

$$\dot{x}^{[p]} = \sum_{i=p-1}^{\infty} A_i^px^{[i]} + uB_i^px^{[i]} \quad ,$$

$$y = \sum_{i=0}^{\infty} C_ix^{[i]} + uD_ix^{[i]} \quad , \tag{2.8}$$

$$x(0) = (1,0,0,\dots\ ) \quad \text{and} \quad |u| \leqslant 1 \quad ,$$

of the type studied by Brockett and Fuhrmann [5]. If (2.8) is truncated by setting $x^{[p]} = 0$ for $p \geqslant q$ the result is a finite dimensional bilinear system

$$\dot{x}^{[p]} = \sum_{i=p-1}^{q-1} A_i^p x^{[i]} + uB_i^p x^{[i]} ,$$

$$y = \sum_{i=0}^{q-1} C_i x^{[i]} + uD_i x^{[i]} \tag{2.9}$$

where $x = x^{[0]}, x^{[1]}, \ldots, x^{[q-1]}$,

$$x(0) = (1,0,\ldots,0) \quad \text{and} \quad |u| \leqslant 1 \quad ,$$

which is used in the proof of the following.

Theorem 1 (Bilinearization around a point): Consider the nonlinear control system

$$\dot{x} = f_o(x) + uf_1(x) \quad ,$$

$$y = g_o(x) + ug_1(x) \quad ,$$

$$x(0) = 0 \quad \text{and} \quad |u| \leqslant 1 \quad ,$$

where $x = (x_1,\ldots,x_n)$, $y = (y_1,\ldots,y_k)$ and u is a scalar. For any integer $q \geqslant 0$ there exists a bilinear control system

$$\dot{x} = Ax + uBx$$

$$y = Cx + uDx$$

$$x(0) = (1,0,\ldots,0) \quad \text{and} \quad |u| \leqslant 1$$

where $x = (x_o,\ldots,x_m)$ $(m \geqslant n)$, $y = (y_1,\ldots,y_k)$ and u a scalar such that for some constants M, $T > 0$ and for any admissible control $u(t)$ the outputs $y(t)$ and $\overline{y}(t)$ of the nonlinear and bilinear system satisfy

$$|y(t) - \overline{y}(t)| \leqslant Mt^q \quad , \quad \forall\, t \in [0,T] \quad .$$

Moreover if $x(t)$ is the state of the nonlinear system and $\overline{x}(t)$ consists of x_1 to x_n of the state of the bilinear system then

$$|x(t) - \overline{x}(t)| \leqslant Mt^{q+1} , \quad \forall\, t \in [0,T] \quad .$$

The proof is based on the following.

Lemma: Consider the two control systems

$$\text{(i)} \quad \dot{z} = h_1(z,u) \quad \text{and} \quad \text{(ii)} \quad \dot{z} = h_2(z,u)$$

where $|u| \leqslant 1$, $z = (z_o,\ldots,z_m)$ and $z(0) = (1,0,\ldots,0)$.

Suppose there exists constants K_1 and $T_1 > 0$ and an integer $q \geqslant 0$ such that for any solution $z(t)$ of (i) where $0 \leqslant t \leqslant T_1$,

$$|h_1(z(t),u(t)) - h_2(z(t),u(t))| \leqslant K_1 t^q \quad .$$

Furthermore suppose there exists constants K_2, $\varepsilon > 0$ such that for any

$u, z, \bar{z}$ where $|u| \leq 1$, $|z| \leq \varepsilon$, $|\bar{z}| \leq \varepsilon$

$$|h_2(z,u)-h_2(\bar{z},u)| \leq K_2|z-\bar{z}| \ .$$

Then there exist M, $T > 0$ such that if $z(t)$ and $\bar{z}(t)$ are the solution of (i) and (ii) for the same $u(t)$ then

$$|z(t)-\bar{z}(t)| \leq Mt^{q+1} \ , \quad \forall \ \ 0 \leq t \leq T$$

Proof of Lemma: Choose $0 < T \leq T_1$ such that any solution $z(t)$ and $\bar{z}(t)$ of (i) and (ii) satisfy $|x(t)| \leq \varepsilon$, $|\bar{x}(t)| \leq \varepsilon$ for $0 \leq t \leq T$.

$$|z(t)-\bar{z}(t)| \leq \int_0^t |h_1(z(s),u(s))-h_2(\bar{z}(s),u(s))|ds$$

$$\leq \int_0^t |h_1(z(s),u(s))-h_2(z(s),u(s))|ds$$

$$+ \int_0^t |h_2(z(s),u(s))-h_2(\bar{z}(s),u(s))|ds$$

$$|z(t)-\bar{z}(t)| \leq K_1t^{q+1} + K_2 \int_0^t |z(s)-\bar{z}(s)|ds$$

Choose $M_1 \geq K_2$ such that $M_1^{q+2} \geq (q+1)!K_1$. Let

$$\phi(t) = \int_0^t |z(s)-\bar{z}(s)|ds$$

and

$$\psi(t) = \exp(Mt) - \sum_{j=0}^{q+1} \frac{(M_1t)^j}{j!} = \sum_{j=q+2}^{\infty} \frac{(M_1t)^j}{j!}$$

Then $\phi(t)$ and $\psi(t)$ satisfy

$$\phi'(t) \leq \frac{M_1^{\ q+2}}{(q+1)!} \ t^{q+1} + M_1\phi(t)$$

$$\psi'(t) = \frac{M_1^{\ q+2}}{(q+1)!} \ t^{q+1} + M_1\psi(t)$$

so by a standard comparison theorem [6, p. 25] $\phi(t) \leq \psi(t)$ therefore

$$|z(t)-z(t)| = \phi'(t) \leq \frac{M_1^{\ q+2}}{(q+1)!} \ t^{q+1} + M_1\psi(t)$$

$$\leq M_t^{q+1} \qquad \text{for some } M.$$

Proof of Theorem: Let $x = (x_1,\ldots,x_n)$ and $x^{[p]}$ be defined as before. If $\dot{x} = f(x,u)$ then for any p there exists a function $h_1^p(x,u)$ such that $\dot{x}^{[p]} = h_1^p(x,u)$. Let $(z_o,\ldots,z_m) = (x^{[0]},\ldots,x^{[q-1]})$ then

$$\dot{z} = h_1(z,u) \quad .$$

Let $h_2(z,u)$ be the finite dimensional bilinear system obtained in (2.9).

It is straightforward to verify that these systems satisfy the hypotheses of the lemma. The theorem follows immediately.

3. LINEARIZATION AND BILINEARIZATION AROUND A REFERENCE TRAJECTORY

Consider the nonlinear control system

$$\begin{aligned} \dot{x} &= f_o(t,x) + uf_1(t,x) \\ y &= g(t,x) \end{aligned} \tag{3.1}$$

where x is an n vector, y a k vector and u a scalar. The case of vector controls is a straightforward generalization. Time appears explicitly in (3.1) because we intend to consider perturbations around a reference trajectory which we conveniently take to be $x(t) \equiv 0$ generated by the control $u(t) \equiv 0$ for $0 \leqslant t \leqslant T$. expanding the right side of (3.1) as before we obtain a time dependent linear system,

$$\begin{aligned} \dot{x} &= A(t)x + uB(t) \\ y &= C(t)x \end{aligned} \tag{3.2}$$

where $A(t) = \partial f_o/\partial x\ (t,0)$, $B(t) = \partial f_1/\partial u(t,0)$ and $C(t) = \partial g/\partial x\ (t,0)$. It is well-known (for example Halkin [6]) that the error of approximating (3.1) by (3.2) is proportioned to the L' norm of the control. That is there exists an M such that if x(t) and y(t) are the solutions of (3.1) for some u(t) and $\overline{x}(t)$ and $\overline{y}(t)$ are the corresponding solutions of (3.2) u(t) then

$$|x(t)-\overline{x}(t)| \leqslant M\varepsilon^{\lambda}$$

and

$$|y(t)-\overline{y}(t)| \leqslant M\varepsilon^{\lambda}$$

where

$$\varepsilon = \int_0^T |u(t)|dt$$

is sufficiently small.

For second order approximations consider

$$\begin{aligned} \dot{x} &= \frac{\partial f_o}{\partial x}\ (t,0)x + x^T\ \frac{\partial^2 f_o}{\partial x^2}\ (t,0)x + uf(t,0) + u\ \frac{\partial f}{\partial x}\ (t,0)x, \\ y &= \frac{\partial g}{\partial x}\ (t,0) + x^T\ \frac{\partial^2 g}{\partial x^2}\ (t,0)x\ . \end{aligned} \tag{3.3}$$

Using Carleman's technique again we obtain a time varying bilinear system

$$\begin{aligned} x^{[0]} &\equiv 1, \qquad x^{[p]} \equiv 0 \qquad \text{for } p \geqslant 3, \\ \dot{x}^{[1]} &= A^1_1(t)x^{[1]} + A^1_2(t)x^{[2]} + uB^1_o(t)x^{[0]} + uB^1_1(t)x^{[1]}\ , \\ \dot{x}^{[2]} &= A^2_2(t)x^{[2]} + uB^2_1(t)x^{[1]}\ , \\ y &= C_1(t)x^{[1]} + C_2(t)x^{[2]} \end{aligned}$$

Notice that as in (3.2) terms involving u can be truncated sooner than those with x only without reducing the accuracy of the approximation.

In general one obtains an infinite dimensional time varying bilinear system

$$\begin{aligned} \dot{x}^{[p]} &= \sum_{i=p}^{\infty} A_i^p(t)x^{[i]} + uB_{i-1}^p(t)x^{[i-1]} \\ y &= \sum_{i=1}^{\infty} C_i(t)x^{[i]} \end{aligned} \tag{3.5}$$

which when truncated yields a finite dimensional time varying bilinear system

$$\begin{aligned} \dot{x}^{[p]} &= \sum_{i=p}^{q-1} A_i^p(t)x^{[i]} + uB_{i-1}^p(t)x^{[i-1]} \\ y &= \sum_{i=1}^{q-1} C_i(t)x^{[i]} \quad . \end{aligned} \tag{3.6}$$

Theorem (Bilinearization around a reference trajectory): Consider the nonlinear control system

$$\begin{aligned} \dot{x} &= f_o(t,x) + uf_1(t,x) \\ y &= g(t,x) \end{aligned}$$

where $x = (x_1,\ldots,x_n)$, $y = (y_1,\ldots,y_k)$, u a scalar, and the reference trajectory $x(t) \equiv 0$ is generated by $u(t) \equiv 0$ for $0 \leqslant t \leqslant T$. For any $q \geqslant 0$ there exists a bilinear control system

$$\begin{aligned} \dot{x} &= A(t)x + uB(t)x \\ y &= C(t)x \end{aligned}$$

where $x = (x_o,\ldots,x_m)$ $(m \geqslant n)$, $y = (y_1,\ldots,y_k)$ and u a scalar such that for some M any control $u(t)$ the outputs $y(t)$ and $\overline{y}(t)$ of the nonlinear and bilinear system satisfy

$$|y(t)-\overline{y}(t)| \leqslant M\varepsilon^q \ , \qquad \text{for } 0 \leqslant t \leqslant T \ ,$$

provided

$$\varepsilon = \int_0^T |u(t)|dt$$

is sufficiently small. Moreover if $x(t)$ is the state of the nonlinear system and $x(t)$ is coordinater x_1 through x_m of the state of the bilinear system then

$$|x(t)-\overline{x}(t)| \leqslant M\varepsilon^q \ , \qquad \text{for } 0 \leqslant t \leqslant T \ .$$

Once again the proof follows from a lemma.

Lemma: Consider the two systems

$$\text{(i)} \quad \dot{z} = h_1(t,z,u) \qquad\qquad \text{(ii)} \quad \dot{z} = h_2(t,z,u)$$

where $z = (z_o,\ldots,z_m)$, u is a scalar and the reference trajectory $z(t) \equiv (1,0,\ldots,0)$ is generated by $u(t) \equiv 0$, for $0 \leqslant t \leqslant T$, for both systems. Suppose there exists an integer $q \geqslant 0$ and a constant K_1 such that for any solution $z(t)$ of (i)

$$|h_1(t,z(t),u(t))-h_2(t,z(t),u(t))| \leqslant K_1(|z(t)|^q+|z(t)|^{q-1}|u(t)|)$$

Suppose there exists $\delta > 0$ and a constant K_2 such that for any u, $|z| \leq \delta$, $|\overline{z}| \leq \delta$, $0 \leq t \leq T$

$$|h_2(t,z,u)-h_2(t,\overline{z},u)| \leq K_2|z-\overline{z}|$$

Finally suppose there exists a constant K_3 such that for any u $|z| \leq \delta$ and $0 \leq t \leq T$

$$|h_1(s,z,u)| \leq K_3|u| \quad \text{and} \quad |h_2(s,z,u)| \leq K_3|u| \quad .$$

If $z(t)$ and $\overline{z}(t)$ are the solution of (i) and (ii) for some $u(t)$ then

$$|z(t)-\overline{z}(t)| \leq M\varepsilon^q \ , \qquad 0 \leq t \leq T$$

if $\varepsilon = \int_0^T |u(t)|dt$ is sufficiently small.

Proof of Lemma: If $z(t)$ and $\overline{z}(t)$ are the solutions of (i) and (ii) for some $u(t)$ then

$$|z(t)| \leq \int_0^t |h_1(s,z(s),u(s))|ds \leq K_3 \int_0^t |u(s)|ds \leq K_3\varepsilon$$

$$|\overline{z}(t)| \leq \int_0^t |h_2(s,\overline{z}(s),u(s))|ds \leq K_3\int_0^t |u(s)|ds \leq K_3\varepsilon$$

Restricted to ε small enough so that $|z(t)| \leq \delta$ and $|\overline{z}(t)| \leq \delta$, for $0 \leq t \leq T$,

$$\begin{aligned} |z(t)-\overline{z}(t)| &\leq \int_0^t |h_1(s,z(s),u(s))-h_2(s,z(s),u(s))|ds \\ &\quad + \int_0^t |h_2(s,z(s),u(s))-h_2(s,\overline{z}(s),u(s))|ds \\ &\leq K_1 \int_0^t |z(s)|^q+|z(s)|^{q-1}|u(s)|ds+K_2\int_0^t |z(s)-\overline{z}(s)|ds \end{aligned}$$

$$|z(t)-\overline{z}(t)| \leq K_1(T+1)\varepsilon^q+K_2\int_0^t |z(s)-\overline{z}(s)|ds \quad .$$

Let

$$\phi(t) = \int_0^t |z(s)-\overline{z}(s)|ds$$

and

$$\psi(t) = \frac{K_1(T+1)\varepsilon^q}{K_2}\exp(K_2t) - \frac{K_1(T+1)\varepsilon^q}{K_2}$$

Then $\phi(t)$ and $\psi(t)$ satisfy

$$\phi'(t) \leq K_1(T+1)\varepsilon^q + K_2\phi(t)$$

$$\psi'(t) = K_1(T+1)\varepsilon^q + K_2\psi(t)$$

so once again

$$|z(t)-\overline{z}(t)| = \phi'(t) \leq \psi'(t) \leq M\varepsilon^q$$

for some M and all t, $0 \leq t \leq T$.

Proof of Theorem: Let $x = (x_1,\ldots,x_n)$ and $x^{[p]}$ be defined as before. If $\dot{x} = f_o(t,x)+uf_1(t,x)$ then for any p let $h_1^p(x,u)$ be the function such that $\dot{x}^{[p]} = h_1^p(x,u)$. Let $(z_o,\ldots,z_m) = (x^{[0]},\ldots,x^{[q-1]})$ then $\dot{z} = h_1(z,u)$.

Let $h_2(z,u)$ denote the right side of finite dimensional time varying bilinear system (3.6). It is straightfoward to verify that h_1 and h_2 satisfy the hypothesis of the lemma.

4. CONCLUSION

These two theorems indicate that in a local sense around a point or reference trajectory the class of bilinear systems is dense in the class of nonlinear systems with control entering linearly. One views this with somewhat mixed feelings, on the one hand it implies that nonlinear behavior can be effectively approximated by bilinear models. On the other hand it implies that complexity of bilinear systems is almost as great as that of nonlinear systems and that theoretical results for the whole class of bilinear systems will be almost as difficult as for the class of nonlinear systems.

The above methods of bilinearizing around a point and reference trajectory are not the only ways of doing this. One advantage of these methods is that the nonlinear approximation to the state of the nonlinear system is easily read from the state of bilinear systems. However the bilinear system will generally not be of smallest state dimension among those bilinear systems which approximate the nonlinear system with a desired order of accuracy. For an alternate approach to bilinearizing around a point we refer the reader to [7].

Besides more accurate quantitative information that higher order bilinearization afford they also increase our qualitative knowledge of the system. For example, by linearizing around a reference trajectory one is able to deduce only first order necessary conditions for optimality, i.e. the Pontryagin maximal principle. By using essentially the quadratic bilinearization Kelley, Kopp and Mayer [8] were able to develop new necessary conditions, not implied by the Maximal Principle. From this we see that higher order bilinearizations yield new qualitative information about the boundary of the set of accessible points of a control system.

The error analysis of the above theorems did not assume any stability properties for the nonlinear systems being approximated. In many applications these systems do have some form of stability. Further research is needed to discover how these improve the approximations.

The above bilinearizations were accomplished using monomials in the state variables to approximate the nonlinear behavior of the system. It is possible (see [1] and [2]) to use other families of functions and the question arises when would it be advantageous to do so.

REFERENCES

1. Carleman, T., "Application de la théorie des équations intégrales singulières aux équations differentialles de la dynamique," T. Ark. Mat. Astrom. Fys. 22B,1 (1932).
2. Bellman, R. and J.M. Richardson, "On some questions arising in the approximate solution of nonlinear differential equations," Quart. of App. Math. 20, 4, (1963) p. 333-339.

3. Brockett, R.W., "On the Algebraic Structure of Bilinear Systems," in Variable Structure Control Systems, (R. Mohler and A. Ruberti, eds.) Academic Press, N.Y., 1972, pp. 153-168.
4. Brockett, R.W., "Lie algebras and Lie groups in control theory," in Geometric Methods in System Theory (D.O. Mayne and R.W. Brockett, eds.) D. Riedel, Boston, 1973, pp. 43-82.
5. Brockett, R.W. and P.A. Fuhrmann, "Normal Symmetric Dynamical Systems," to appear.
6. Halkin, H., "Mathematical Foundations of System Optimization," in Topics in Optimization (G. Leitmann, ed.) Academic Press, N.Y., 1967, pp. 197-262.
7. Krener, A.J., "Bilinear and Nonlinear Realizations of Input-output Maps," SIAM J. Control, to appear.
8. Kelley, H.J., R.E. Kopp and H.G. Moyer, "Singular Extremals," in Topics in Optimization (G. Leitmann, ed) Academic Press, N.Y., 1967, pp. 63-101.

CONTROLLABILITY AND ACCESSIBILITY CRITERIA FOR BILINEAR SYSTEMS*

DAVID L. ELLIOTT
Washington University
St. Louis, Missouri

ABSTRACT

The most important criterion for controllability or accessibility of a homogeneous bilinear system is the transitivity of the associated Lie algebra. This report on work in progress shows how to use Boothby's list of transitive Lie algebras to obtain explicit conditions for transitivity, and a theorem on controllability is given.

INTRODUCTION

The controllability problem for the bilinear system

$$\dot{x} = (A+u_t B)x \tag{1}$$

is usually studied on the state-space $R^n - \{0\} = R_0^n$, with (for example) controls u piecewise-constant and $|u(t)| \leq 1$. The most important necessary condition for controllability is due to Kučera [1,2] and can be put in the following form:

Let L be the smallest real linear space of matrices that contains A and B and is closed under the Lie product $[X,Y] = XY-YX$; let $\{C_1,\ldots,C_m\}$ be a basis of L; if system (1) is controllable, then L is transitive, i.e.,

$$\text{rank } (C_1x,\ldots,C_mx) = n \text{ for all } x \in R_0^n . \tag{2}$$

This condition is sufficient [1] for controllability of the "symmetric" system

$$\dot{x} = (v_t A + u_t B)x ; \tag{3}$$

i.e., for any x' and x" in R_0^n there exists a time $T > 0$ and a control such that if $x(0) = x'$ then $x(T) = x''$ for the corresponding solution of the system (3). The condition (2) is also sufficient for the system (1) to have the accessibility property, which means that for every x' in R_0^n the set of states x", which can be reached at some time from x', has open interiors; Sussmann and Jurdevic [3] proved that.

The difficulty with condition (2) is it requires that no real non-zero vector x should satisfy $\binom{m}{n}$ simultaneous polynomial equations of degree n. In theory, the Tarski-Seidenberg Theorem could be applied and Cheng [4] has shown that the problem reduces to testing quadratic equations, but the difficulty remains.

However, the space L is a Lie algebra generated by A and B, and the controllability problem has a Lie group formulation [5]. Using this formulation Boothby [6] has obtained a list of all the Lie algebras (in canonical form) that satisfy condition (2). Thus the problem becomes somewhat easier: one computes enough information about L to decide

---

*This research was supported by the National Science Foundation through Grant GK-38694.

whether it belongs to this list of transitive Lie algebras. It is the author's hope that useful controllability criteria can be obtained in this way. The present paper presents what little progress has been made toward that goal, a special result for the case n=2, and a controllability theorem for the case that B has rank one.

TRANSITIVE LIE ALGEBRAS

For the full list of transitive algebras, the reader should see Boothby's paper [6]; J.L. Sedwick's forthcoming dissertation [7] gives more details about their matrix forms. The list below covers the cases n odd ($n \neq 7$), and n even but not a multiple of 4. d_i is the dimension of L_i as a linear space.

Let $\mathcal{C}_0 = \{0\}$, $\mathcal{C}_1 = \{\lambda I,\ \lambda \in R\}$, $\mathcal{C}_2$ will be defined. These are the possible centers, i.e. $\mathcal{C} = \{X \in L\colon\ [X,Y]=0\ \forall Y \in L\}$.

If n is odd ($n \neq 7$), the only transitive Lie algebras L are:

(L_1) $\mathfrak{gl}(n,R) = \{\text{all real } n\times n \text{ matrices } X\}$; $d_1 = n^2$;

(L_2) $\mathfrak{sl}(n,R) = \{X\colon\ \operatorname{tr} X=0\}$; $d_2 = n^2 - 1$;

(L_3) $\mathfrak{so}(n,R) \oplus \mathcal{C}_1 = \{X\colon\ X=Y+\lambda I,\ Y' = -Y,\ \lambda \in R\}$

since $\mathfrak{so}(n,R) = \{Y\colon\ Y' = -Y\}$; $d_3 = 1+\frac{1}{2}(n^2-n)$.

The Lie algebra L_3 is given in "canonical form", that is, L_3 represents a set of algebras which can be reduced to this form by a similarity transformation PXP^{-1}. The same is true of L_4, L_5 and L_6 below.

There is a well-known correspondence of complex numbers with matrices: $x+iy \overset{\alpha}{\sim} \begin{pmatrix} x & y \\ -y & x \end{pmatrix}$. Analogously we define, for real $m\times m$ matrices, the representation

$$\alpha(X+iY) = \begin{pmatrix} X & Y \\ -Y & X \end{pmatrix};$$

$$\text{let } \mathcal{C}_2 = \alpha(\lambda I_m + \mu i I_m) = \begin{pmatrix} \lambda I_m & \mu I_m \\ -\mu I_m & \lambda I_m \end{pmatrix}.$$

If n=2m and m is odd, the only transitive Lie algebras are L_1, L_2, L_3, and

(L_4) $\mathfrak{su}(m) \oplus \mathcal{C}_2$, where $\mathfrak{su}(m)$ is the class of matrices of form

$$\begin{pmatrix} X & Y \\ -Y' & Z \end{pmatrix} \quad \begin{array}{l} X,\ Y,\ Z \text{ real } m\times m \text{ matrices,} \\ X \text{ and } Z \text{ skew-symmetric;} \\ d_4 = 2m^2 - m = 2 + \frac{1}{2}(n^2-n) \end{array}$$

(L_5) $\mathfrak{sl}(m,C) \oplus \mathcal{C}$, $\mathcal{C}=\mathcal{C}_0$, $\mathcal{C}_1$, $\alpha(\mu i I_m)$, or $\mathcal{C}_2$,

$\mathfrak{sl}(m,C) = \{\alpha(X+iY),\ \operatorname{tr} X=0=\operatorname{tr} Y\}$,

where X, Y are $m\times m$ real matrices;

$\dim \mathfrak{sl}(m,C) = \frac{n^2}{2} - 2$, $d_6 = \frac{n^2}{2} - 2 + d_c$, $d_c = 0,1,1,$ or 2;

(L_6) $\mathcal{sp}(m,R)\oplus\mathcal{C}$, $\mathcal{C} = \mathcal{C}_0$ or $\mathcal{C}_1$,

$$\mathcal{sp}(m,r) = \left\{ \begin{pmatrix} X & Y \\ Z & -X' \end{pmatrix}, \quad \begin{matrix} Y' = Y \\ Z' = Z \end{matrix} \right\},$$

X, Y, Z real $m\times m$; $d_6 = \frac{1}{2}(n^2+n)$ or $1 + \frac{1}{2}(n^2+n)$.

To compute the Lie algebra L generated by A and B, one can use the fact that $L = \text{span } \{F_i\}_0^{n^2-1}$, where $F_0 = \{A,B\}$ and $F_{i+1} = \{[A,X], [B,X]\colon X \in F_i\}$, truncating this tree of matrices whenever we obtain a commutator $[A,X]$ or $[B,X]$ which lies in the (real linear) span of those previously obtained. The tree is finite, since there are at most n^2 linearly independent $n\times n$ matrices.

It can also be seen easily that A and B can be replaced by any two independent linear combinations $\alpha_1 A + \beta_1 B$, $\alpha_2 A + \beta_2 B$, in computing L, and that for fixed non-singular P they can be replaced by PAP^{-1}, PBP^{-1}, if either operation is helpful.

Our restriction to single-input bilinear systems is only for convenience, since the situation for $\dot{x} = (A+u^1B_1+ \ldots +u^rB_r)x$ is no different; Boothby [6] shows that every transitive Lie algebra generated by r+1 matrices can also be generated by two matrices.

Assuming that n is not 7 or divisible by 4, we can decide the transitivity of L by testing whether it belongs to the list (L_1-L_6). The following facts are helpful, and decisive for n odd.

(a) If $\dim L=n^2$, $L=L_1$.

(b) If $\text{tr } A = \text{tr } B = 0$ and $\dim L = n^2-1$, then $L=L_2$.

(c) Let L_0 be the Lie algebra generated by $A_0=A-(\text{tr}A)I$, $B_0=B-(\text{tr}B)I$. If A_0 and B_0 satisfy $A_0'H + HA_0 = 0$, $B_0'H + HB_0 = 0$ for some positive definite symmetric H, L_0 has a skew-symmetric canonical form. If also $\dim L_0 = \frac{1}{2}(n^2-n)$, then $L_0 = \mathcal{so}(n,r)$. If also either trA or trB is non-zero, $L=L_3$.

Apparently the even-dimensional cases L_4, L_5 and L_6 (as well as those not mentioned for n a multiple of 4) can be similarly checked. If n=4, the only transitive Lie algebra not on the list (L_1-L_6) is the standard 4-dimensional representation of the quaternions, which is easy to test for.

If L is determined not to be on the list, it is not transitive. This possibility can also be checked in other ways. Thus, if A and B can (with the same transformation P) both be put in upper-triangular form, then L is a solvable Lie algebra and not transitive. There are tests for A and B to have a common invariant subspace, due to Brockett [8], which guarantee non-controllability. Also, if L is compact then it is not transitive (on R_0^n); it suffices for compactness that both A and B be skew-symmetric (compare (c) above).

There are other sufficient conditions for transitivity and controllability, not involving Boothby's list. First we define the linear operator adA by $\text{adA}(X) = [A,X]$; $(\text{adA})^2(X) = [A,[A,X]])$, etc.

Theorem: If in system (1) B has rank one, so $B=bc'$, and (A,b), $(A'c)$ are controllable pairs, then

$$\text{rank } \{Bx, [A,B]x, \ldots, (\text{adA})^{n^2-1}(B)x\} = n \text{ for all } x \neq 0 \qquad (4)$$

(this is stronger than transitivity); if in addition there exists positive definite symmetric H such that $A'H + HA = 0$, then (1) is controllable.

Proof: The matrix in (4) is $\{b(c'x), Ab(c'x)-b(c'Ax),\ldots\}$; if its rank were less than n then $c'x=0$, $c'Ax=0,\ldots$ from the full rank of $(b,Ab,\ldots)$; but from the full rank of $(c,A'c,\ldots)$, $x=0$ (cf. Brockett [9]). The remaining conclusion is proved by Cheng [4,10].

THE TWO-DIMENSIONAL CASE, AND CONCLUSION

For 2×2 matrices A,B complete results can be obtained. Since two-dimensional systems are frequently encountered in applications and as theoretical examples, the details will be set forth. To begin with, suppose A and B are linearly independent, otherwise L is one-dimensional.

Without loss of generality we can assume one of the matrices, say A, has trace zero and that A is in one of the real canonical forms

$$A_0 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad A_1 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix},$$

by suitable choice of α_1, α_2, and P. The Lie algebra generated by A_0 and B contains $\mathfrak{sl}(2,R)$, and is transitive if the component $B_{21}\neq 0$ in B. In case A_1, L contains $\mathfrak{sl}(2,R)$, if $B_{12}B_{21}\neq 0$. In either case L is $\mathfrak{gl}(2,R)$ if also $\mathrm{tr}B\neq 0$. In case A_2, L is transitive if $B+B'\neq 0$; L is of L_3 type if B is scalar, otherwise L_2 or L_1 type.

The only work necessary to prove all that is to compute [A,B] and check the list, which for n=2 degenerates to L_1, L_2, L_3.

In general, determining transitivity, like chemical qualitative analysis, requires some flow-charting. In addition, checking spaces of matrices, viewed as n^2-dimensional vectors, for dimension is not trivial. However, the difficulty is so much less than that of applying (2) directly, for $n > 2$, that the transitivity problem can now be considered solvable.

I wish to acknowledge the help of Gen-Sen Cheng, Jack Sedwick, and T.J. Tarn in developing these ideas.

REFERENCES

[1] J. Kučera, "Solution in Large of Control Problem $\dot{x}=(A(1-u)+Bu)x$," Czech. Math. J 16 (1966), pp. 600-622.

[2] J. Kučera, "Solution in Large of Control Problem $\dot{x}=(Au+Bv)x$," Czech. Math J. 17 (1967), pp. 91-96.

[3] H.J. Sussmann and V. Jurdjevic, "Controllability of Nonlinear Systems," J. Diff. Equations 12 (1972), pp. 95-116.

[4] G.S.J. Cheng, "Controllability of Discrete and Continuous-Time Bilinear Systems," D.Sc. Dissertation, Washington University, June 1974.

[5] D.L. Elliott and T.J. Tarn, "Controllability and Observability for Bilinear Systems," SIAM National Meeting, Seattle, 1971; Report CS-722, Washington University.

[6] W.M. Boothby, "A Transitivity Problem from Control Theory," J. Diff. Equations, to appear.

[7] J.L. Sedwick, "The Equivalence of Nonlinear and Bilinear Control Systems," D.Sc. Dissertation, Washington University, to appear.

[8] R.W. Brockett, "On the Algebraic Structure of Bilinear Systems," in Variable Structure Control Systems, (R. Mohler and A. Ruberti, eds.) Academic Press, 1972, pp. 153-168.

[9] R.W. Brockett, "Lie Algebras and Lie Groups in Control Theory," in Geometric Methods in System Theory, (D.Q. Mayne and R.W. Brockett, eds.) D. Reidel Publishing Co., Dordrecht, 1973, pp. 43-82.

[10] G.S.J. Cheng, D.L. Elliott and T.J. Tarn, "Controllability of Bilinear Systems," in Proc. Second U.S.-Italy Seminar on Variable-Structure Systems (R. Mohler and A. Ruberti, eds.), to be published in Springer Lecture Notes.

ANALYSIS OF BILINEAR STOCHASTIC SYSTEMS

ALAN S. WILLSKY*, STEVEN I. MARCUS**, AND DAVID N. MARTIN***
Electronic Systems Laboratory
Department of Electrical Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

ABSTRACT

In this paper we consider the analysis of stochastic dynamical systems that involve multiplicative (bilinear) noise processes. After defining the systems of interest, we consider the evolution of the moments of such systems, the question of stochastic stability, and estimation for bilinear stochastic systems. Both exact and approximate methods of analysis are introduced, and, in particular, the uses of Lie-theoretic concepts and harmonic analysis are discussed.

I. INTRODUCTION

Recently, a great deal of effort has gone into developing a detailed theory for a class of nonlinear systems that possesses a great deal of structure -- the class of bilinear systems. The mathematical tools behind bilinear system analysis include not only many of the vector space techniques that are so valuable in linear system theory but also a number of tools drawn from the theories of Lie groups and differential geometry. It is the purpose of this paper to describe the several mathematical techniques that have been developed for the analysis of bilinear stochastic systems. The reader is referred to [1]-[7] for detailed descriptions of some of these techniques and for further references.

II. STOCHASTIC BILINEAR SYSTEMS

We consider the dynamical model

$$\dot{x}(t) = [A_0 + \sum_{i=1}^{N} u_i(t)\, A_i]x(t) \tag{1}$$

where the A_i are given n x n matrices, the u_i are scalar inputs, and x is either an n-vector or an n x n matrix. Let $\mathscr{L}$ be the Lie algebra generated by $A_0,\ldots,A_n$ -- i.e. the smallest subspace of n x n matrices containing $A_0,\ldots,A_n$ and closed under the commutator product $[P,Q] = PQ-QP$--and let G be the Lie group associated with $\mathscr{L}$ - - i.e. the smallest group (under matrix multiplication) containing $\exp(L)\ \forall\ L \in \mathscr{L}$. It is well known that for piecewise continuous u_i, we have $x(t) \in G\,x(0)$, t. If the u_i are white noise processes, we have that $x \in Gx(0)$ holds almost surely for the solution to the Ito equation

$$dx(t) = \{[A_0 + \frac{1}{2}\sum_{i,j=1}^{N} R_{ij}(t)A_iA_j]dt + \sum_{i=1}^{N} A_i dv_i(t)\}\, x(t) \tag{2}$$

where v is a Brownian motion with $E[dv(t)dv'(t)] = R(t)dt$. If u is smoother than white noise - - for instance if u satisfies

---

* Assistant Professor of Electrical Engineering, M.I.T. The work of this author was supported by NSF under grant GK-42090 and by NASA under grant NGL-22-009-124.

** National Science Foundation Fellow.

*** Presently with Raytheon Company, Bedford, Mass.

$$d\xi(t) = F(t)\xi(t)dt + G(t)dw(t) + \alpha(t)dt, \quad u(t) = H(t)\xi(t) \tag{3}$$

where w is a Brownian motion -- we needn't include the correction term contained in (2). In this case x is not a Markov process, but the pair (x,ξ) is; however, when we regard (x,ξ) as the state, the system (1),(3) involves products of state variables. As we shall see, the analysis in this case is more difficult than in the white noise case.

III. MOMENT EVOLUTION EQUATIONS

3.1 Systems Driven by White Noise

In this subsection we present moment equations derived by Brockett [4]. Consider (2) where, for simplicity, we assume that $R(t) = I$. Recall that the number of linearly independent homogeneous polynomials of degree p in n variables (i.e. $f(cx_1,\ldots,cx_n) = c^p f(x_1,\ldots,x_n)$) is

$$N(n,p) = \begin{pmatrix} n+p-1 \\ p \end{pmatrix} \tag{4}$$

We choose a basis for this space of polynomials consisting of the elements

$$\sqrt{\begin{pmatrix} p \\ p_1 \end{pmatrix}\begin{pmatrix} p-p_1 \\ p_2 \end{pmatrix}\cdots\begin{pmatrix} p-p_1-\cdots-p_{n-1} \\ p_n \end{pmatrix}}\; x_1^{p_1} \cdots x_n^{p_n} \;;\; \sum_{i=1}^{n} p_i = p,\; p_i \geq 0 \tag{5}$$

If we denote the vector of these elements (ordered lexicographically) by $x^{[p]}$, then $||x||^p = ||x^{[p]}||$, where $||x||^2 = x'x$. It is clear that if x satisfies $\dot{x} = Ax$, then $x^{[p]}$ satisfies $\dot{x}^{[p]} = A^{[p]}x^{[p]}$. We regard this as the definition of $A^{[p]}$, which is closely related to Kronecker sum matrices [10],[11].

The Stratonovich analog of (2) is

$$dx(t) = [A_0 dt + \sum_{i=1}^{N} A_i dv_i(t)]x(t) \tag{6}$$

Since Stratonovich equations obey the rules of ordinary calculus, we obtain

$$dx^{[p]}(t) = \left[A_0^{[p]} dt + \sum_{i=1}^{N} A_i^{[p]} dv_i(t) \right] x^{[p]}(t) \tag{7}$$

Using the moment equations for Stratonovich equations, we have an evolution for the pth moments of (2):

$$\frac{d}{dt} E(x^{[p]}(t)) = \{A_0^{[p]} + \frac{1}{2} \sum_{i=1}^{N} (A_i^{[p]})^2\} E(x^{[p]}(t)) \tag{8}$$

3.2 Systems Driven by Colored Noise: Exact Expressions

Definition 1: We associate with any Lie algebra $\mathscr{L}$ its derived series

$$\mathscr{L}^{(0)} = \mathscr{L}$$

$$\mathscr{L}^{(n+1)} = [\mathscr{L}^{(n)},\mathscr{L}^{(n)}] = \{[A,B] \mid A,B \in \mathscr{L}^{(n)}\}, \; n \geq 0 \tag{9}$$

$\mathscr{L}$ is abelian if $\mathscr{L}^{(1)} = \{0\}$ and is solvable if $\mathscr{L}^{(n)} = \{0\}$ for some n.

Theorem 1: A matrix Lie algebra $\mathscr{L}$ is solvable if and only if there exists a nonsingular matrix $\mathbf{P}$ (possibly complex-valued) such that PAP^{-1} is upper triangular (zero below diagonal) for all $A \varepsilon \mathscr{L}$.

Proof: See [13].

We now consider (1) where we assume that u is a zero mean Gaussian random process with $E[u(t)u'(s)] = \mathbf{P}(t,s)$. Let $\mathscr{L}$ be the Lie algebra generated by $A_0, A_1, \ldots, A_N$. We claim that one can construct closed form expressions for $E[x^{[p]}(t)]$ if $\mathscr{L}$ is solvable. Assuming that L is solvable (and that $x(0)$ is independent of u), we can obtain a closed form expression for Φ_u, the state transition matrix for (1), as a function of u. First we find the matrix $\mathbf{P}$ as in Theorem 1 such that each $B_i = PA_iP^{-1}$ is upper triangular. Then we can solve the equation

$$\dot{\Psi}_u(t,0) = [B_0 + \sum_{i=1}^{N} B_i u_i(t)]\Psi_u(t,0), \quad \Psi_u(0,0) = I \tag{10}$$

by straightforward calculations (see the example in Subsection 4.2), and we compute

$$x(t) = \mathbf{P}^{-1}\Psi_u(t,0)\mathbf{P}\,\mathbf{x}(0) \tag{11}$$

and Ψ_u involves nothing more complicated than exponentials of integrals of components of u, polynomials in u, and various combinations, products, and integrals of such quantities. Since u is Gaussian, we can evaluate the expectations of such quantities in closed form (see Subsection 4.2) and can thus obtain a closed form expression for $E[x(t)]$. We can also obtain closed form expressions for $E[x^{[p]}(t)]$, as $x^{[p]}$ consists of the same types of functionals of u (see [7]).

3.3 Systems Driven by Colored Noise--Approximate Analysis

Consider the system (1) where u is given by (3). In this section we describe an approximate method that can be used for moment analysis if L is not solvable. This approach is based on the truncation of the cummulants of a random process [14] and can be used for arbitrary nonlinear systems, although it is particularly well-suited to systems with polynomial nonlinearities. We will illustrate the approach by example and refer the reader to [1], [2], and [9] for more detailed discussions.

Consider the scalar example

$$dx(t) = \alpha x^2(t)dt + \beta x(t)dw(t) \tag{12}$$

where $E[dw^2(t)] = dt$. We compute [12]

$$\dot{m}_1(t) = \alpha m_2(t), \quad \dot{m}_2 = 2\alpha m_3(t) + \beta^2 m_2(t)$$
$$\dot{m}_3(t) = 3\alpha m_4(t) + 3\beta^2 m_3(t) \tag{13}$$

where $m_i = E(x^i)$.

From these equations it is clear that any attempt to study the properties of (13) must involve a truncation of the infinite set of coupled equations.

As discussed in [1] and [14], the direct truncation method -- setting to zero all moments greater than some given order -- can cause difficulties, but the truncation of the cumulants -- the coefficients of the Taylor series expansion of the logarithm of the characteristic function -- is quite reasonable (for instance, only the first two cumulants are nonzero for

Gaussian densities). By comparing terms in the series expansions for the characteristic function, one can compute relationships between the cumulants and moments. For instance, the fourth cumulant satisfies

$$k_4 = m_4 - 3m_2^2 - 4m_1m_3 + 12m_1^2m_2 - 6m_1^4 \tag{14}$$

Thus if we assume $k_4 = 0$ and use (14) to express m_4 in terms of m_i, $i \leq 3$, we can use this approximation in the moment equations (13).

Finally, we note that the cumulants technique can be used to design suboptimal estimators for bilinear systems (see [2])

IV. STOCHASTIC STABILITY

A problem of considerable theoretical and practical interest is the question of the stability of linear systems containing multiplicative noise processes. As discussed in [7] and [8], such systems often arise in linear feedback systems in which the actuator, sensor, and feedback gains may be stochastic in nature. In this case, a model such as (1) arises, in which the u_i represent the random elements in the system.

Definition 2: A vector random process x is pth order stable if $E[x^{[p]}(t)]$ is bounded for all t, and x is pth order asymptotically stable if

$$\lim_{t\to\infty} E[x^{[p]}(t)] = 0 \tag{15}$$

The systems (1),(3) and (2) are pth order (asymptotically) stable if the solution x is pth order (asymptotically) stable for all initial conditions x(0) that are independent of ξ (in the (1),(3) case) or the v_i (in (2)) and such that $E[x^{[p]}(0)] < \infty$.

4.1 The White Noise Case

Consider the system (2). Using (8), we have the following.

Theorem 2: The system (2) with R=I is pth order asymptotically stable if and only if the matrix

$$D_p = A_0^{[p]} + \frac{1}{2}\sum_{i=1}^{N} (A_i^{[p]})^2 \tag{16}$$

has all its eigenvalues in the left-half plane ($\mathrm{Re}(\lambda) < 0$). The system is pth order stable if D_p has all its eigenvalues in $\mathrm{Re}(\lambda) \leq 0$ and if the eigenvalues with real parts = 0 have all higher order residues = 0 (i.e. there are no $te^{\lambda t}$ terms in $e^{D_p t}$ with $\mathrm{Re}(\lambda) = 0$; see [15, p.55]).

For examples of this result, we refer the reader to [7].

4.2. The Colored Noise Case

We have seen in Subsection 3.2 that we can obtain closed form moment expressions for systems of the form given in (1) when u is a Gaussian colored noise process and $\mathscr{L}$ is solvable. We can use these expressions to obtain stochastic stability criteria. We illustrate these ideas by means of an example (see [7] for other examples).

Example 1: Consider the system

$$\dot{x}(t) = [A_0 + A_1\xi_1(t) + A_2\xi_2(t)]x(t) \tag{17}$$

$$A_0 = \alpha \begin{bmatrix} 2 & -1 \\ 1 & 0 \end{bmatrix}, \quad A_1 = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 1 & -1 \\ 0 & 0 \end{bmatrix} \tag{18}$$

with ξ_1, ξ_2 zero mean and

$$E\left[\xi_i(t)\,\xi_i(t+\tau)\right] = \frac{1}{2}\left[\sigma_1^2 e^{-k_1|\tau|} + \sigma_2^2 e^{-k_2|\tau|}\right], \quad i=1,2 \tag{19}$$

$$E\left[\xi_1(t)\,\xi_2(t+\tau)\right] = \frac{1}{2}\left[\sigma_1^2 e^{-k_1|\tau|} - \sigma_2^2 e^{-k_2|\tau|}\right]. \tag{20}$$

In this case L is solvable, and defining $y = \mathbf{P}x$, with

$$\mathbf{P} = \begin{bmatrix} 0 & 1 \\ 1 & -1 \end{bmatrix} \tag{21}$$

we have

$$y(t) = \begin{bmatrix} \eta_1(t,0) & \int_0^t \eta_1(t,s)\eta_2(s,0)[\alpha + \xi_1(s)]ds \\ 0 & \eta_2(t,0) \end{bmatrix} \tag{22}$$

$$\eta_i(t_1,t_2) = \exp \ \alpha(t_2-t_1) + \int_{t_1}^{t_2} \xi_i(s)ds \tag{23}$$

Some lengthy but straightforward calculations yield the following necessary and sufficient condition for first order asymptotic stability

$$\alpha < -\frac{1}{2}\left(\frac{\sigma_1^2}{k_1} + \frac{\sigma_2^2}{k_2}\right) \tag{24}$$

and the system is first order stable if equality holds in (24).

V. ESTIMATION FOR BILINEAR SYSTEMS

There have been a number of results obtained on estimation for bilinear systems. In this section we illustrate two classes of results. The first of these involves estimation on nilpotent Lie groups. For such systems one can obtain finite dimensional optimal nonlinear filters [3], and we illustrate the result for the 3-dimensional case in Subsection 5.1. An approximate technique based on the use of harmonic analysis on compact Lie groups is illustrated in Subsection 5.2. For more detailed discussions of bilinear estimation, we refer the reader to [2],[3],[5], and [6].

5.1 Estimation on Nilpotent Lie Groups

Definition 3: A Lie algebra L is called nilpotent if the sequence of Lie algebras

$$\mathscr{L}^0 = \mathscr{L}$$

$$L^{n+1} = [L, L^n] = \{[A,B] \mid A \in L, B \in L^n\}, \quad n \geq 0 \tag{25}$$

terminates in zero. Note that abelian => nilpotent => solvable, but none of the reverse implications hold in general.

Consider the system (1) with $A_0 = 0$, $N=3$, and

$$A_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad A_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \tag{26}$$

It is easy to see that these matrices generate a nilpotent Lie algebra. Also, suppose that u satisfies (3) with (for simplicity) $H = I$ and $\alpha = 0$, and let x be the 3x3 transition matrix for (1) (i.e. $x(0) = I$). Then

$$x(t) = \begin{bmatrix} 1 & \int_0^t \xi_1(\sigma)d\sigma & \int_0^t \xi_2(\sigma)d\sigma + \int_0^t \int_0^{\sigma_1} \xi_1(\sigma)\xi_3(\sigma_2)d\sigma_2 d\sigma_1 \\ 0 & 1 & \int_0^t \xi_3(\sigma)d\sigma \\ 0 & 0 & 1 \end{bmatrix} \tag{27}$$

Suppose we have the linear observations

$$dz(t) = C(t)\xi(t)dt + R^{1/2}(t)dv(t) \tag{28}$$

where v is a Brownian motion process and $R > 0$. **Then the conditional mean, $\hat{x}(t|t) = E(x(t)|z(s), 0 \leq s \leq t)$, can be generated by a finite** dimensional nonlinear filter as follows (see [2],[3] for details): let

$$P(\sigma_1,\sigma_2,t) = E[(\xi(\sigma_1) - \hat{\xi}(\sigma_1|t))(\xi(\sigma_2) - \hat{\xi}(\sigma_2|t))'|z(s), 0\leq s\leq t] \tag{29}$$

Then $P(\sigma,t,t) = k(t,\sigma)\Sigma(t)$, **and is nonrandom, where**

$$\dot{\Sigma} = F\Sigma + \Sigma F' + GG' - \Sigma C'R^{-1}C\Sigma \tag{30}$$

$$K'(t,\sigma) = -[F'(t) + \Sigma^{-1}(t)G(t)G'(t)]K'(t,\sigma) \ ; \quad K(\sigma,\sigma) = I \tag{31}$$

Consider the additional state variables

$$\dot{\eta}(t) = \xi(t) \tag{32}$$

$$\dot{\alpha}(t) = \left[\int_0^t K_3'(t,\sigma)e_1' d\sigma \right] \xi(t) - [F'(t) + \Sigma^{-1}(t)G(t)G'(t)]\alpha(t) \tag{33}$$

$$\dot{\beta}(t) = [e_1 e_3']\eta(t) - [F'(t) + \Sigma^{-1}(t)G(t)G'(t)]\beta(t) \tag{34}$$

where $\eta(0) = \alpha(0) = \beta(0) = 0$, K_j is the jth row of K, and e_i is the ith unit vector in R^3. We implement a Kalman filter for (3), (28), (32)-(34), and then

$$\hat{x}(t|t) = \begin{bmatrix} 1 & \hat{\eta}_1(t|t) & \hat{\eta}_2(t|t) + \hat{\gamma}(t|t) \\ 0 & 1 & \hat{\eta}_3(t|t) \\ 0 & 0 & 1 \end{bmatrix} \tag{35}$$

where $\hat{\gamma}(0|0) = 0$ and

$$\begin{aligned} d\hat{\gamma}(t|t) = {} & \left[\hat{\xi}_1(t|t)\hat{\eta}_3(t|t) + \int_0^t \Sigma_{13}(t,\sigma,t)\,d\sigma \right] dt \\ & + [\hat{\alpha}'(t|t)+\hat{\beta}(t|t)]\, \Sigma(t)C'(t)R^{-1}(t)[dz(t)-C(t)\hat{\xi}(t|t)dt] \end{aligned} \tag{36}$$

The key to this result is the fact that **K** is a state transition matrix with a finite-dimensional representation (31). Also, as mentioned earlier, this result can be extended to arbitrary nilpotent Lie groups, some solvable Lie groups, and to other nonlinear systems. The key to these extensions is the fact that products and transposes of weighting patterns with finite-dimensional realizations also have finite-dimensional realizations (see [3]).

5.2 The Use of Harmonic Analysis in Suboptimal Filter Design

In this subsection we illustrate the use of harmonic analysis in bilinear estimation by studying a phase tracking problem with the aid of Fourier series analysis. For extensions of these results to other problems, including attitude estimation with the aid of spherical harmonic analysis, we refer the reader to [2]. Suppose we receive the signal z defined by

$$dz(t) = [\sin\theta(t)]dt + r^{1/2}(t)dw(t) \tag{37}$$

$$d\theta(t) = w_c dt + q^{1/2}(t)dv(t) \tag{38}$$

where v and w are independent standard Brownian motions independent of the random initial phase $\theta(0)$. We desire to track the signal phase, θ mod 2π. As discussed in [6], a useful estimation criterion is to choose $\tilde{\theta}(t)$ to minimize $E[1-\cos(\theta(t)-\tilde{\theta}(t))|z(s),0 \leq s \leq t]$.

Putting the problem into Cartesian coordinates by defining $x_1 = \sin\theta$, $x_2 = \cos\theta$, we have the bilinear equations

$$\begin{bmatrix} dx_1(t) \\ dx_2(t) \end{bmatrix} = \begin{bmatrix} -q(t)dt/2 & (w_c dt+q^{1/2}(t)dv(t)) \\ -(w_c dt+q^{1/2}(t)dv(t)) & -q(t)dt/2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \tag{39}$$

$$dz(t) = x_1(t)dt + r^{1/2}(t)dw(t) \tag{40}$$

and the optimal estimate is

$$\tilde{\theta}(t) = \tan^{-1} \frac{\hat{x}_1(t|t)}{\hat{x}_2(t|t)} \tag{41}$$

We note that $\tilde{x}_1 = \sin\tilde{\theta}$, $\tilde{x}_2 = \cos\tilde{\theta}$ are minimum variance estimates of x_1 and

x_2 under the constraint $\tilde{x}_1^2 + \tilde{x}_2^2 = 1$.

To solve this problem, we expand the conditional probability density for $\theta(t)$ given $\{z(s),\ 0 \leq s \leq t\}$ in a Fourier series. The coefficients are

$$c_n(t) = \frac{1}{2\pi} E[e^{-in\theta(t)} | z(s),\ 0 \leq s \leq t] = b_n(t) - ia_n(t) \tag{42}$$

We then have the coupled equations [6]

$$\begin{aligned} dc_n &= -[inw_c + \frac{n^2}{2} q]c_n dt \\ &+ \frac{(c_{n-1} - c_{n+1})}{2i} + 2\pi c_n I_m(c_1) \left[\frac{dz + 2\pi I_m(c_1)dt}{r}\right] \end{aligned} \tag{43}$$

See [6] for a discussion of the structure of the optimal filter. Note that $x_2 - ix_1 = 2\pi c_1$, and thus we need only compute c_1 to determine $\tilde{\theta}$; however, c_2 couples into c_1, c_3 into c_2, etc. Thus we need to approximate the optimal filter to obtain an implementable system. One approach considered in [6] is to make an approximation based on the assumption that the conditional density is a "folded normal density," the circular analog of the normal density. The coefficients of such a density depend on only two parameters

$$2\pi c_n = e^{-n^2\gamma/2} e^{-in\eta} \tag{44}$$

and $c_2 = 8\pi^3 |c_1|^2 c_1^2$. Using this relationship as an approximation in (44), we obtain an equation for c_1 that is uncoupled from the other coefficients. We refer the reader to [6] for simulation results that indicate that this filter performs better than the optimal phase-lock loop. In addition, directly analogous results using spherical harmonics have been obtained in [2] for the estimation of processes on the sphere.

VI. CONCLUSIONS

In this paper we have described a number of techniques for the analysis of bilinear stochastic systems. As can be seen from our results and from those in the references, this is a class of systems that is rich in both structure and practical applications.

References

1. A.S. Willsky and S.I. Marcus, "Analysis of Bilinear Noise Models in Circuits and Devices," Monograph of the Colloquium on the Application of Lie Group Theory to Nonlinear Network Problems, 1974 IEEE International Symposium on Circuits and Systems, San Francisco, Calif., April 1974.

2. A.S. Willsky and S.I. Marcus, "Estimation for Bilinear Stochastic Systems," presented at the U.S.-Italy Seminar on Variable Structure Systems, Portland, Oregon, May 26-31, 1974; also M.I.T. Electronic Systems Laboratory Report ESL-R-544.

3. S.I. Marcus and A.S. Willsky, "A Class of Finite-Dimensional Optimal Nonlinear Filters", Fifth Symposium on Nonlinear Estimation and Its Applications, San Diego, Calif., Sept., 23-25, 1974.

4. R.W. Brockett, "Lie Theory and Control Systems on Spheres," SIAM J. Appl. Math., Vol. 25, No. 2, Sept. 1973, pp.213-225.

5. A.S. Willsky, Dynamical Systems Defined on Groups: Structural Properties and Estimation, Ph.D. Thesis, Dept. of Aeronautics and Astronautics, M.I.T., Cambridge, Mass., June 1973.

6. A.S. Willsky, "Fourier Series and Estimation on the Circle with Applications to Synchronous Communication I: Analysis," and "II: Implementation," IEEE Trans. on Inf. Theory, Vol. IT-20, No. 5, Sept. 1974.

7. D.N. Martin, S.I. Marcus, A.S. Willsky, and T.L. Johnson, "On the Stochastic Stability of Linear Systems Containing Colored Multiplicative Noise," to be submitted for publication.

8. D.N. Martin, Stability Criteria for Systems with Colored Multiplicative Noise, Ph.D. Thesis, Dept. of Electrical Engineering, M.I.T., Cambridge, Mass., June 1974.

9. T. Nakamizo, "On the State Estimation for Non-Linear Dynamic Systems," Int. J. Control, Vol. 11, No.4, 1970, pp.683-695.

10. R.W. Brockett, "Lie Algebras and Lie Groups in Control Theory," in Geometric Methods in System Theory, D.Q. Mayne and R.W. Brockett, eds., Reidel Pub. Co., The Netherlands, 1973.

11. R. Bellman, Introduction to Matrix Analysis, McGraw-Hill, New York, 1965.

12. A.H. Jazwinski, Stochastic Processes and Filtering Theory, Academic Press, New York, 1970.

13. A.A. Sagle and R.E. Walde, Introduction to Lie Groups and Lie Algebras, Academic Press, New York, 1973.

14. R.L. Stratonovich, Topics in the Theory of Random Noise, Volume I, transl. by R.A. Silverman, Gordon and Breach, New York, 1963.

15. R.W. Brockett, Finite Dimensional Linear Systems, John Wiley and Sons, New York, 1970.

TRACKING RANDOMLY MOVING OBJECTS WITH A COMPUTER VISION SYSTEM

VINCENT C. JONES
University of Illinois, Urbana, Illinois

The needs and requirements of computer vision are discussed, with particular emphasis on the need for the ability to handle moving objects by next generation robots. The problems encountered due to different types of motion are considered, and a method for providing a robot with real-time vision processing of moving objects is presented. The utility of the method is illustrated by a robot parts picker being developed. It can survey a workspace containing numerous objects moving in random fashion, identify the objects while keeping track of their position and orientation, and retrieve any desired object on command.

INTRODUCTION

The problems encountered in using robot vision in a dynamic environment are widely varied. Robots must be able to handle an object moving in the field of view in order to handle just about any dynamic problem. There are three fundamental situations of interest. In the first case, a stationary robot looks at a moving object. Even this case is not trivial. Since the object´s location is continuously changing, by the time the object is identified the robot may no longer know where it is. In addition, the object may be changing orientation or appearence; only an extremely limited class of objects looks identical whether viewed from the front, side, or back (e.g., spheres). Furthermore, in the real world, the object´s appearance may actually be changing, and confusion due to shadow lines or relative motion of related parts must be avoided. A horse is still a horse whether running or nibbling grass.

If the robot is moving and the rest of the world is not, the problem is very different. Here it is undesirable for the robot to require a fixed amount of computer time to determine its current location and the location of objects around it each time it makes a move. But in current systems, after the robot moves, which it does blindly, it must stop and take a completely fresh look at the world, developing a new world model from ground zero. The robot´s world may have changed so drastically that previous knowledge of the world is almost worthless until enough recognition has occurred with the latest view to make correlation possible. This problem of a moving visual scene is not restricted to mobile robots. Any vision system which is not totally fixed presents this problem. Current systems [2] must use sophisticated software and highly calibrated hardware just to pan the camera over a scene.

Research supported by the Joint Services Electronics Program (U.S. Army, U.S. Navy and U.S. Air Force) under contract number DAAB-07-72-C-0259.

Hence, the usual solution [1][3][4] has been to make sure that every possible point of interest is within the camera's view, and only look at portions of the picture. While this method insures easy calibration, it usually results in tremendous waste of picture resolution. Similarly, a zoom lens is extremely difficult to accurately calibrate. Obtaining the actual size of an object is quite difficult, so rather than calibrating the hardware to determine the size of an object, the opposite approach is normally used, and an object whose size is known is used to calibrate the zoom lens. Unfortunately, this method is prone to disaster if the program confuses the identification of two objects of similar appearance but different size.

Finally, there is the problem of the moving robot in a dynamic environment. Here, not only is the robot's field of view constantly changing, but objects may move within the field of view. If the robot cannot keep track of significant objects in real time, the situation is virtually hopeless.

A ROBOT WITH DYNAMIC VISION

A robot "parts picker" with real time hand-eye coordination, able to select a desired part from an unstructered flow of parts in its field of vision is being developed. It is able to recognize the desired part, and not be confused by general similarities. This means that the recognition algorithm is not fast enough to keep up with parts motion directly. What is done instead is to use a separate tracker, working with the recognizer, to keep track of the location, motion, and identification of objects. The tracker can also select portions of the field of view to pass on to the recognizer. The recognizer, working along with the tracker, does not have to do an instantaneous job of object recognition but instead works on objects one at a time, asking the tracker for more information or new views as required. Once an object is recognized, the tracker flags it, and gives the recognizer another object to work on. The tracker is only required to distinguish an object from the background, a vastly simpler (and consequently faster) problem than differentiating between numerous objects.

THE OBJECT TRACKER

The object tracker provides the means to achieve real time vision processing. The tracker must be able to locate and track objects with a minumum of knowlege. When an object first enters the field of view, it must be tracked using only features detectable without explicit knowlege of the actual object. The object must be tracked using only this minimal information until the recognizer has time to look at it. Since the tracker may be following several objects simutaneously, it is vital that the computational overhead for each object be as small as possible. The tracker must not be fooled by changes in the appearance of the object. For example, if an object is approaching the robot, its identity does not change even though its size does, while a wrench looks very different when viewed from the top rather than the side. However, while it must recognize gross enough details not to be confused by changes in size or orientation, it should differentiate well enough not to be confused by similar objects near one another. The trade-off between processing speed and recognition ability is of key importance.

The tracker is based on the routine FINDIT, which directs the "hard core" searching of raw picture data. FINDIT chooses from a repetoire of procedures those most likely to detect the desired object based on information contained in a "tracking data block" for the object. The tracker takes a picture, then passes a pointer to the picture and a pointer to the particular tracking data block to FINDIT. The first word of the data block is a flag word. Each bit, if a one, means its corresponding word is valid. If this word is flagged as invalid, FINDIT immediately fails. The object ID is then checked (if it is flagged as valid). Objects can belong to general "search type" classes (e.g., part of robot - physical position is exact, light simple object, dark simple object, complex object - use special description block, etc.). The predicted position of the object (in picture coordinates) is then calculated using last known position, velocity and acceleration (robot part positions are transformed directly from known real world coordinates).

A thirtysix cell square window is then extracted from the picture, centered on the predicted position. The window maps the picture one to one for small objects, every other point on every other line for large objects. All searching is limited to these fixed size windows, enabling the use of fast, efficient search routines. In the case of uniform intensity objects, a simple, very fast region grower uses available, object and background intensity data to expand a region about the predicted position. Only if this fails is the whole window scanned for an appropriate region. If the region found does not contradict any of the known properties of the desired region, it is considered the desired region. Here is where speed becomes critical. The prime criterion for identification is location, which for randomly moving objects can be considered predictable only on a short time scale.

The tracker has been implemented on the PDP-10 computer at the Coordinated Science Lab using the CSL vision system. It can follow light and dark objects on a plain background. Moderate variations in object or background intensity are ignored, as long as there is contrast between the background and the object. Looking at a scene once a second, the tracker can handle over twenty distinct objects. Tracking a single object, it can take and process over ten pictures a second, and only loses the object when it "disappears" in the vidicon's lag; i.e., when its image becomes so blurred that there is no distinct boundary between the object and the background.

THE OBJECT RECOGNIZER

The object recognizer is not required to obtain every last bit of information from each individual peek at the world. In fact, it cannot even get a complete view of the world. Instead, it looks at one object at a time, passed on to it from the tracker. This leads to simplifications, since confusion between objects is eliminated. The location of each object in view is maintained by the tracker, so it is not necessary for the recognizer to complete its work before another frame is taken. However, the recognizer cannot be too slow or the system might wind up tracking too many unidentified objects and miss "seeing" the desired goal before it moves out of view.

The recognizer can also utilize information from later frames in its analysis. Because the recognizer does not have to be concerned with the entire field of view, it does not have to recognize every object in a scene before it decides to concentrate on a particular object. The recognizer can, if necessary, keep trying different algorithms to identify an object, and can use such "non-visual" information as speed, acceleration, position, orientation etc., provided by the tracker, to help in its decision making process.

REFERENCES

[1] Minsky, M. & Papert, S. "Proposal to ARPA for Research on Artificial Intelligence at MIT," MIT AI Memo 245, October 1971.

[2] Sobel, Irwin "On Calibrating Computer Controlled Cameras for Perceiving 3-D Scenes,"Advance Papers of the Third International Joint Conference on Artificial Intelligence, Stanford Univ., August 1973.

[3] Yakimovsky, Y. & Feldman, J. "A Semantics-Based Decision Theory Region Analyzer,"Advance Papers of the Third International Joint Conference on Artificial Intelligence, Stanford Univ., August 1973.

[4] "News Roundup," Modern Data, Dec 1972

FEATURE EXTRACTION FROM COLOR CONTRAST IN VISUAL SCENES

STEPHEN A. UNDERWOOD
Department of Electrical Engineering
The University of Texas at Austin
Austin, Texas 78712

J. K. AGGARWAL
Department of Electrical Engineering
The University of Texas at Austin
Austin, Texas 78712

Summary

One of the difficult problems in visual scene analysis is extracting the individual features from complex visual scenes. One application where color is a defining factor of a feature is in the use of color infrared film to detect vegetation in aerial photographs. Vegetation is a high reflector of the infrared spectrum and shows a high color contrast on the photograph. This color contrast is more detectable than the density contrast in black and white infrared photographs. The types and conditions of vegetation can also be determined from the color.

We are presently using a three-color film scanner system to examine aerial color infrared photographs of citrus orchards. The color contrast is used to locate the individual trees in the photograph. This is accomplished by scanning the photograph and locating individual "blobs" of homogenous red areas which conform to certain constraints. Overlaping adjacent trees can be separated by algorithms which consider the circular structure of the trees. Since we are concerned with orchards, the row structure of the trees can also be used as a factor in eliminating unwanted data from the photograph.

Once the individual trees have been located, the color density of each tree can be used to determine various types of information about the tree. The objective is to detect very small insect infestations in localized areas of orchards so that localized spraying can be used instead of large scale spraying after the infestation has spread.

This research was sponsored by Joint Services Electronics Program and by National Science Foundation under Grant No. GK-42790.

AUTOMATIC VISUAL INSPECTION OF HYBRID CIRCUITS

WESLEY E. SNYDER
Coordinated Science Laboratory, University of Illinois, Urbana

ABSTRACT

A computer program has been written which detects defects in hybrid circuits by processing a television image of the circuit. The program is in two phases, a "learning" phase and an "inspecting" phase. The learning phase uses interactive graphics to learn the physical structure of the circuit. The output of the learning phase is the inspecting phase. This phase is also a computer program and sets up data and control structures which model the circuit in terms of component models, themselves computer programs. The design to the system is based upon this concept of using procedures as models for the objects being viewed. Use is also made of edge detection, region growing, and determination of color as phases of component identification and verification.

INTRODUCTION

Hybrid circuits are miniature circuits in which the resistors and conductors are placed on a (usually) ceramic substrate by a printing process. Other components, including capacitors, discrete transistors, and integrated circuits may be attached by glue or a solder process. A computer program has been written to inspect these ciruits for defects. Some typical defects are missing components, components with their surface damaged by foreign matter (eg., a blob of solder), or mislocated components.

MOTIVATION

In computer programming, there is a distinction between "data structures" and "control structures". Classically, the data structures which contain information stored in the program are distinguished from the control structures which contain the procedures by which the information is accessed. Winograd's[2] work in natural language was an attempt to incorporate the information of the program into its control structure by using the syntactic patterns and semantic relationships in English sentences. In his program, the definition of a word is a procedure to be executed when the parser encounters that word rather than a dictionary listing.

We follow a somewhat related approach in the area of computer vision. A model for an object is a procedure to be executed when the object is encountered, not a static data base entry. In this case, recognition does not consist of a search for the model in the data base which best matches the object in

---

Research supported by the Joint Services Electronics Program (U. S. Army, U.S. Navy, and U. S. Air Force) under contract number DAAB-07-72-C-0259.

the scene, Instead the program executes a model or series of models to determine whether the object encountered matches the criteria for recognition.

In this "procedural model" approach any model can call another model and/or return hints as to which model to call next. This gives the program a high degree of flexibility.

The inspection of hybrid circuits provides a context in which to examine problems relating to the procedural models concept such as formal representation of procedural models, effective implementation of such models, and appropriate programming languages in which to deal with this type of control/data structure.

Another consideration in the development of this system was its applicability. The inspection system was developed with an attempt to make use of realistic assembly line conditions.

While hybrid circuits are quite complex and noisy (visually), they are largely two dimensional. Their analysis is thus more tractible than analysis of a three dimensional scene of equivalent complexity.

Let us now review briefly the operation of the system.

OPERATION

The system consists of a "learning" phase and an "inspection" phase, where the latter is the output of the former. In the learning phase, an operator presents the system with an assortment of pictures of the circuit and uses interactive graphics to "teach" the system about the physical arrangement of the components. As the system "learns", it builds a model for the entire circuit, based upon models of individual components.

This circuit model is a computer program which, when compiled and executed, becomes the inspection phase.

The component models are presently hand coded and vary considerably, depending on the properties of the object being viewed. Resistors, for example, can be characterized by the fact that they are very dark and (usually) have straight edges. Wires, on the other hand, are three dimensional in that they obscure other objects, they can be bent in a variety of ways, and can possess both highlights (direct reflections of the light source) and nulls (reflections of very dark areas). The algorithms used to detect wires are too involved to be explained in detail here, but make use of the reflective properties of the metal and the expected location of the ends of the wires.

Some of the "tools" used by the system include the Hueckel[1] edge detection operator, recursive region growing, and various applications of the properties of color-- intensity, saturation, and hue.

The system is written in BLISS. It runs on a PDP-10 and requires from 35 to 55 k words, depending on how much overlaying is done. A typical inspection requires ten seconds of computer time.

Future work includes an extension of the project to allow the system to relate more abstract, "schematic" descriptions of circuits to their visual images.

BIBLIOGRAPHY

[1] Hueckel, M., "An operator which locates edges in digitized pictures," JACM, Oct. 1973.

[2] Winograd, T. "Procedures as a Representation for Data in a Computer Program for Understanding Natural Language." PhD Thesis, MIT, 1971.

A CONSTRAINT SATISFACTION METHOD FOR INFERENCE MAKING

JOHN GASCHNIG*
Carnegie-Mellon University
Pittsburgh, Pennsylvania

ABSTRACT

A general constraint satisfaction method, based upon the method of Waltz, for solving certain classes of state space problems is presented. The method avoids the computational redundancy of searching by eliminating locally inconsistent assignments as candidates for global search. For sufficiently constrained problems the method is very powerful. Its behavior parallels some types of reasoning by inference used by humans.

INTRODUCTION

State space problems have been studied by artificial intelligence researchers for many years. In a state space problem, there are a given number of variables, a set of discrete a priori possible values for each variable, and a well-defined criteria for testing whether a particular assignment of values to variables is a solution. Consider, for example, the cryptarithmetic problem DONALD + GERALD = ROBERT. The problem is to assign to each of the ten letters a distinct digit so that a valid sum is obtained when the letters are replaced by the digits. A typical approach to the solution of this problem is to use a depth-first search which takes advantage of the local nature of the solution criteria. In the search each new instantiation is tested for consistency with each of the instantiations made higher up in the tree, and backtracking occurs when an inconsistency is discovered. Figure 1 shows a condensed portion of such a search for DONALD + GERALD = ROBERT.

Although this method is much more efficient than a "generate and test" method, there is still much redundancy. During the search it is discovered that if N=6 then B=3, which is inconsistent with the previous assignment R=3 (see figure 1). This inconsistency is independent of the particular assignments to A and E, rather it is due to the assignments made to D, T, L, and thus R. Later, when the assignments to A and E have changed, but those of D, T, L, and R have not, the still operative inconsistency must be explicitly rediscovered. The information previously acquired at one node of the tree is not available for later use in other parts of the tree.

Consider also the possible assignments for E. Because of the column O+E=O, it must be the case that either E=0 or E=9, and this fact is independent of the assignments to all other variables. Yet many subtrees for which E has other assignments will be fruitlessly explored by the search.

Figure 2 illustrates an extremely redundant search. In this example there are five variables, each with two possible values. Suppose that the only inconsistent assignment pairs are (A=1, E=1), (A=1, E=2), (A=2, E=1), and (A=2, E=2). There is no solution, yet the search will examine every node in the tree, even though a more intelligent search mechanism would discern that the circled portions of the tree need not be examined at all.

A more sophisticated reasoning mechanism might avoid these undesirable trial and error pitfalls by means of a sequence of clever inferences. Simon [5] gives an example of such an inference sequence for DONALD+GERALD=ROBERT (assuming that D=5 is given): D=5 implies that T=0 and that the carry out of the rightmost column is one. Hence, R=2L+1 is odd. In the leftmost column, D=5 implies that $R > 5$, and thus R=7 or R=9. From the column O+E=O it must be the case that E=0 or E=9. Since T=0, E=9, and then R=7. Since E=9, A must be 4 and there must be a one carried in from column 2, which means 2L+1 = 17, so L=8. Since E=9, the carryout of column 5 must be 1, regardless of the assignment to O, and thus G=1. N, B, and O are the only remaining unassigned variables, but there are only 3!=6 possibilities left, so a simple trial and error at this point yields N=6, B=3, and O=2.

---

*This work was supported by the Advanced Research Projects Agency of the office of the Secretary of Defense (F44620-73-C-0074) and is monitored by the Air Force Office of Scientific Research. This document has been approved for public release and sale; its distribution is unlimited.

We observe in this inference sequence that:

1) There were no instantiations (and thus no trial and error), except trivially, after the bulk of the problem had been solved by inference alone.
2) The inferences were made on the basis of local information, but each had a global impact.
3) The inference making approach does not in itself guarantee that a complete solution will result, but rather a reduced state space.
4) The power of the inference method must depend upon the degree of constraint present in the problem instance.

The question of interest to us is whether such sophisticated lines of inference can be obtained using an unsophisticated mechanism.

DESCRIPTION OF THE METHOD

We now describe a simple method which is capable of making precisely the same inferences as those made by Simon in the above example. The basic method CS1 described below was invented independently by Fikes [2] and Waltz [6], although they did not describe it abstractly as a general method, nor demonstrate its applicability as such. Waltz' great success in using it for parsing line drawings motivated this work. A generalization of CS1 resulting in a more powerful algorithm, CS2, is described later.

The method takes advantage of the structure of problems such as cryptarithmetic, namely that there is a set of independent constraints among the variables, and that an assignment is a solution if and only if it simultaneously satisfies each of the constraints. Instead of instantiating, the method eliminates locally inconsistent variable assignments as candidates for global search.

More formally, the class of problems to which the method is applicable is characterized (for the case of binary relations) by:

1) A set of K variables
2) A finite non-empty set V of distinct possible values
3) Associated with each variable i a non-empty set of a priori possible values $P_i \subseteq V$
4) Associated with each (unordered) pair of variables i, j, a constraint relation $R_{ij} \subseteq P_i \times P_j$

   (Note: In order to have unique names for the relations, throughout the remainder of the paper we adopt the notational convention that a reference to a relation R_{ij} assumes $i < j$.)
5) The set of all possible assignments $\{A \mid A \in P_1 \times P_2 \times \ldots \times P_K\}$
6) The following solution criteria: an assignment A is a solution if and only if for all $i, j \leq K$, if $i \neq j$ then $(A(i), A(j)) \in R_{ij}$, where A(i) is the value assigned to variable i in assignment A.

Definition 1: Variables i and j are mutually constraining if $R_{ij} \subset P_i \times P_j$, i.e., if R_{ij} is not the identity relation.

Two useful properties follow immediately.

Property 1: If $(v_a, v_b) \notin R_{ij}$ then no assignment A such that $A(i) = v_a$ and $A(j) = v_b$ is a solution.

Definition 2: R_{ij} supports $v_a \in P_i$ if there exists a v_b in P_j such that $(v_a, v_b) \in R_{ij}$.

Property 2: For a given i, if there exists a j such that R_{ij} does not support v_a in P_i, then no assignment A such that A(i) = v_a is a solution.

The above description is easily generalized to allow n-ary constraint relations. Properties 1 and 2 provide the basis for the inference making mechanism.

Algorithm CS1:

Step 1: For each variable i create a list L_i of the current set of possible values, initialized to P_i. A kxk matrix M is used to record which relations are currently active. For each i,j such that i and j are mutually constraining, initialize M(i,j) to 1. All other elements of M are initialized to -1.
Step 2: Choose an i and j such that M(i,j) =1.
Step 3: Remove from L_i every value v_a (possibly none) for which there is no value v_b in L_j such that $(v_a, v_b) \in R_{ij}$, and similarly remove from L_j every v_c such that there is no v_d in L_i such that $(v_d, v_c) \in R_{ij}$.
Step 4: If a value was removed from L_i, then for all $1 \leq a < i$ such that M(a,i) = 0, set M(a,i) = 1, and for all $i < a \leq k$ such that M(i,a) = 0 set M(i,a) = 1. Similarly, if a value was removed from L_j, then for all $1 \leq a < j$ such that M(a,j) = 0, set M(a,j) = 1, and for all $j < a \leq k$ such that M(j,a)=0, set M(j,a) = 1. Then set M(i,j) =0.
Step 5: Perform any problem specific actions which may be applicable.
Step 6: If either A) for some i, L_i is empty (i.e. a contradiction); B) for all i, L_i contains exactly one value (i.e., a solution); or C) there is no i,j such that M(i,j) = 1; then terminate. Otherwise, go to step 2.

CS1 may be easily extended to allow n-ary constraint relations, although in practice, the execution cost of step 3 necessarily increases.

The mechanism is illustrated in the problem depicted in figure 3, in which there are three variables (variable E3 is shown redundantly). The relations are represented as links between consistent values of the variables. The star indicates which relation is currently being processed. In figure 3a, value v15 is removed from L_1 because there is no value of E2 which supports it. In figure 3b v31 and v33 are removed from L_3 and v17 is removed from L_1. Note that v33 would not have been removed had v15 (its only support in R_{13}) not been removed from L_1 in the preceding step. Note also that the removal of v17 from L_1 removes the only support for v26 in R_{12}, as will be discovered in a subsequent step. The process is continued in succeeding figures, terminating with a solution in figure 3i.

If the process terminates under condition C the result is a reduced state space, which may contain no solution, exactly one solution, or several solutions. Figure 4 illustrates these cases.

If the problem has very little constraint, CS1 may not reduce the state space size (SS) at all. The amount of computational effort in CS1 is dependent primarily on the selection algorithm of step 2. The final result, however, is order independent, since the constraints are conjunctive.

For implementation purposes, the constraining relations may be represented in either computable or tabular (i.e., stored) form.

APPLICATIONS OF THE METHOD

In applying CS1 to the cryptarithmetic problem, there are six constraining relations, one for each column. We assume, as before, that D=5 is given. There are both binary and ternary relations, but in the case of ternary relations, when one of the variables in a column is fixed (e.g., D in column 6) we may treat the relation as a binary relation.

There is also the constraint that the values assigned to the letters be distinct. It is possible to represent this constrain within the framework of CS1, but for the sake of brevity these details are omitted here and we shall assume that whenever a variable has but one possible value, that value will be removed from the possibilities lists of the other variables. (Note that this constraint can be implemented in step 5 of CS1.)

The elements of each relation are the possible variable assignments given only the local column constraints. For a column $d1 + d2 = d3$, where neither the carry in nor the carry out is fixed, an assignment will be included in the relation if it satisfies one of the four equations $d1 + d2 = d3$, $d1 + d2 + 1 = d3$, $d1 + d2 = d3 + 10$, or $d1 + d2 + 1 = d3 + 10$. In cases where the carry in or carry out is fixed, the relation is determined using the appropriate subset of this set of equations (implemented in step 5 of CS1).

Figure 5 shows the steps of processing. Since D=5, the constraining relation between D and T derived from column 1 (counting from the right) has only one element and so T=0, and the carry out is fixed at 1. Removing the value 0 from the possibilities lists of the other variables and using a carry in of 1, the assignments in column 2 must satisfy either 2L+1=R or 2L+1=R+10, giving L=1,3,4,6,8 and R=3,7,9, as shown in figure 5b. In figure 5c, using the constraint of column 6, we get R=7,9 and G=1,3,4. The process continues, duplicating Simon's inferences, until the unique solution is found in figure 5i.

For purposes of illustration the constraints which were processed in this example was optimal. In general, the selection algorithm for a class of problems will not be optimal for each instance in the class. Optimization of the selection algorithm is a goal of a study now in progress.

For this highly constrained problem the method was very powerful, finding the unique solution with a minimum of effort. Since this power derives from the continual immediate propagation of the implications of an inference, the addition of a new constraint or the removal of a constraint can drastically affect the outcome. For example, in DONALD+GERALD=ROBERT if it is not given that D=5, CS1 does not reduce the SS by very much. In the problem CROSS+ROADS=DANGER (given R=6) the resulting SS=153,600. But if it is also given that no variable has the value 0, then the unique solution is found.

When a problem is insufficiently constrained, CS1 yields only a reduced state space rather than a solution or a contradiction. However, if an additional constraint could be added and the process continued, an additional reduction in the SS would be achieved. One way to introduce an additional constraint is to instantiate one of the variables. The consequent removal of the other possible values of the variable may deprive support for some of the possible values of other variables. By instantiating each of the values in turn the given problem is replaced with a number of more highly constrained subproblems, which are more easily solved than the given problem.

We illustrate this generalization of the basis method with an example-the Instant Insanity puzzle. There are four cubes, and each face of each cube is colored with one of four possible colors. The object is to stack the cubes so that for each of the four sides of the stack, each of the four cube faces is differently colored. There are thus four variables (corresponding to the four cubes) and 24 possible values for each variable (corresponding to the 24 possible orientations of the cube). The values are 4-tuples giving the colors of the front, right, back, and left faces of the cube in that orientation. The constraint used is that two cubes may not have identically colored sides facing in the same direction. Because of symmetry we may choose one of the blocks to have only three possible values instead of 24. In addition, due to symmetrical coloring, one of the cubes in this particular example has only 16 possible values. The a priori SS is 27,648.

When the constraints among the variables have been processed, the number of possible values for blocks B2, B3, and B4 have been reduced to 14, 13, and 12, respectively, a reduction in the SS by a factor of 4.2 to 6552.

We now instantiate B1 (which still has three possible values) and perform CS1 on the instantiated subproblem. The unique solution to this subproblem is found. After instantiating the second value of B1, CS1 reduces the SS by a factor 11.4 to 192. At this point we instantiate B2, and for each instantiation a contradiction or solution is found almost immediately.

As suggested in the preceding example, an attractive generalization of the basic constraint satisfaction method takes the form of a recursive depth-first coupling of constraint satisfaction and instantiation.

Algorithm CS2:

Step 1: Perform CS1.
Step 2: If the result is not a solution or a contradiction, choose a variable for instantiation which has several possible values and perform CS2 on each of the instantiated subproblems.

Algorithm CS2 has the property of completeness, i.e., it will find every solution to the problem or determine that none exists. This property of CS2 makes it more powerful than CS1.

Several other problems were used as test cases for CS2. A two dimensional packing problem was tested, in which four shapes, each constructed of unit squares, must be arranged to form a 4 by 6 unit rectangle. The possible values are the possible locations for the shapes within the 4 x 6 rectangular area. The a priori SS is 100,220. The binary constraint in this problem is that a placement of two shapes on the rectange is inconsistent if they overlap. CS1 reduces the SS by a factor of 27. Instantiation of one of the variables gives immediate contradictions and solutions.

Another test problem was the Soma Cube puzzle, in which seven block structures, each composed of three or four unit cubes, must be fitted together to form a three unit cube. The initial SS=$5 \cdot 10^{12}$. The given problem is not highly constrained: the initial CS1 reduces the SS by only 4%. However, successive instantiations to a depth of three results in a solution. The reduction of the SS by CS1 exceeded a factor of one million in some cases. The entire problem was solved with less than 50,000 elemental tests (i.e., testing whether (v_a, v_b) is in R_{ij}), with almost half of these occurring in the initial (i.e., uninstantiated) application of CS1. This was the only problem among those presented here which was too large to execute the algorithm by hand. A program was written to execute CS2 on this problem.

The method was also applied to a space planning problem examined by Eastman [1]. In this problem, six machines must be placed within a workroom subject to several constraints. In order to apply CS2 to this problem, the space in the workroom was discretized into a grid of possible locations for the machines.

A variation of CS1 which proposes priorities as well as eliminating inconsistencies was used by Tenenbaum and Barrow for interpretation of visual scenes [4].

CONCLUSIONS

Because of the formal definition of the general class of problems to which it is applicable, and because of its demonstrated power in the examples given, especially in duplicating Simon's reasoning by inference in cryptarithmetic, in our opinion CS2 merits consideration as a general AI method. The criteria for applicability are restrictive, but for those problems which do fit its mold, the power of the method is noteworthy, if the problem is sufficiently constrained.

A mathematical model of CS1 has been constructed and an analytic investigation of its behavior is currently in progress.

REFERENCES

1. Eastman, C., "Preliminary Report on a System for General Space Planning," Communications of the ACM, Vol. 15, No. 2, February 1972.
2. Fikes, R., "REF-ARF: A System for Solving Problems Stated as Procedures," TN-14, Artificial Intelligence Center, Stanford Research Institute, Menlo Park, Calif., September 1969.
3. Montanari, U., "Networks of Constraints: Fundamental Properties and Applications to Picture Processing," Computer Science Dept., Carnegie-Mellon University, Pittsburgh, PA, January 1971.
4. Nilsson, N., et al., "Artificial Intelligence--Research and Applications," progress report, Artificial Intelligence Center, Stanford Research Institute, Menlo Park, Calif., April 1974.
5. Simon, H., Sciences of the Artificial, M.I.T. Press, 1969.
6. Waltz, D., "Generating Semantic Descriptions from Drawings of Scenes with Shadows," A.I. TR-271, Artificial Intelligence Laboratory, M.I.T., Cambridge, Mass., August 1972.

```
D=5   T=0   L=1   R=3   (G< 0)  ⊗
            L=2   R=5   (G=0)  ⊗
            L=3   R=7   A=1   E=2   N=4   B=1  ⊗
                                    N=6   B=3  ⊗
                                    N=8   B=5  ⊗
                                    N=9   B=6   O=4  ⊗
                                                O=8  ⊗
                        A=2   E=4   N=1   B=8   O=6  ⊗
                                                O=9  ⊗
                                    N=6   B=3  ⊗
                                    N=8   B=5  ⊗
                                    N=9   B=6   O=1  ⊗
                                                O=8  ⊗
                        A=4   E=8   N=1   B=8  ⊗
                                    N=2   B=9   O=1  ⊗
                                                O=6  ⊗
                                    N=6   B=3  ⊗
                                    N=9   B=6   O=1  ⊗
                                                O=2  ⊗
                        A=6   E=2   N=1   B=8   O=4  ⊗
                                                O=9  ⊗
                                    N=4   B=1   O=8  ⊗
                                                O=9  ⊗
                                    N=8   B=5  ⊗
                                    N=9   B=6  ⊗
                          .
                          .
                          .
```

Figure 1. Condensed partial search tree for
$$\begin{array}{r} \text{DONALD} \\ \underline{\text{GERALD}} \\ \text{ROBERT} \end{array}$$

⊗ indicates backtrack point.

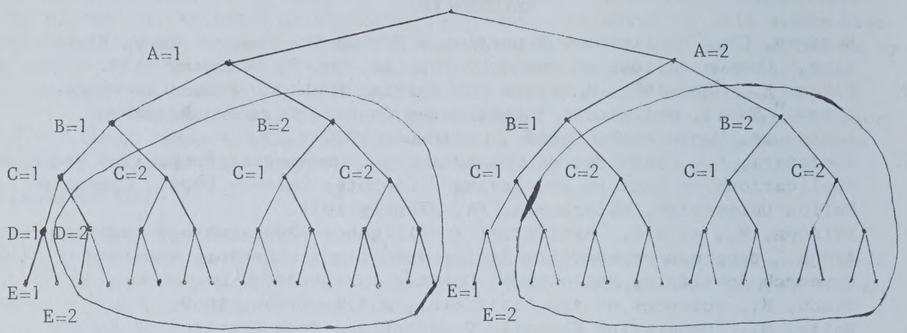

A=1
A=2
B=1
B=2
B=1
B=2
C=1
C=2
C=1
C=2
C=1
C=2
C=1
C=2
D=1
D=2
E=1
E=2
E=1
E=2

Figure 2. A highly redundant search.

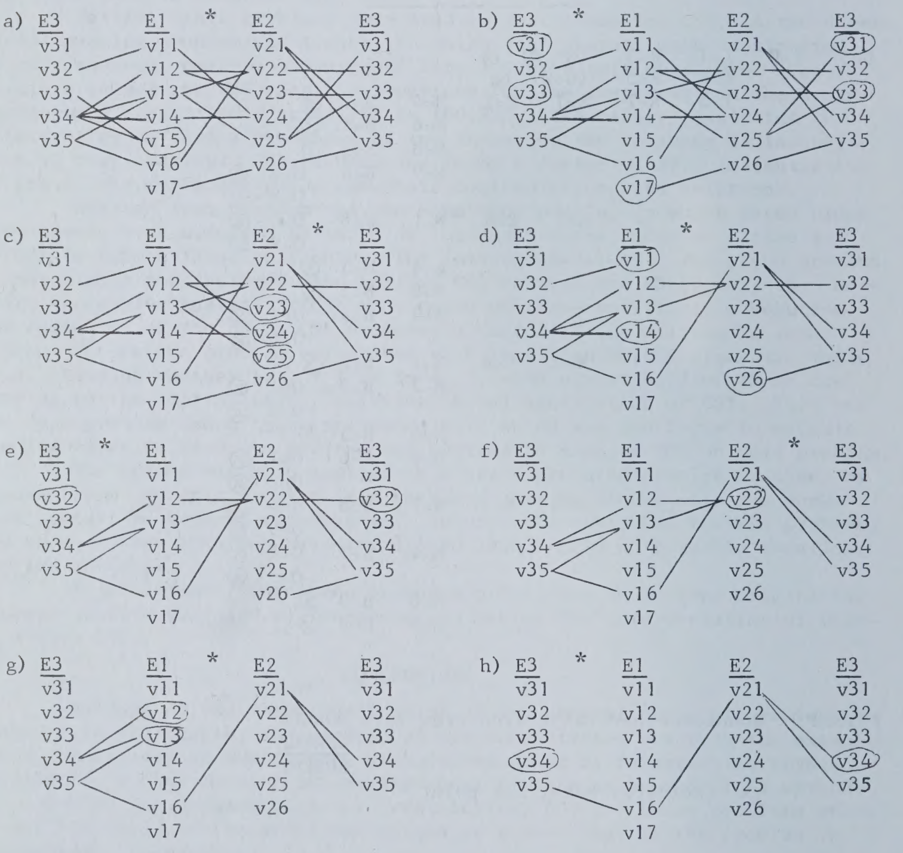

a)
b)
c)
d)
e)
f)
g)
h)
*
E3
E1
E2
E3
v31
v32
v33
v34
v35
v11
v12
v13
v14
v15
v16
v17
v21
v22
v23
v24
v25
v26

Figure 3. Application of CS1 to a sample problem. (Fig. 3 continued on following page.) * indicates which relation is being processed. Circled values are removed.

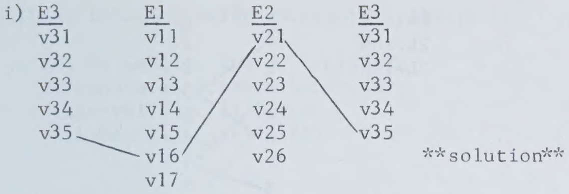
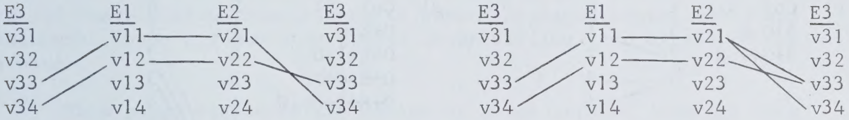
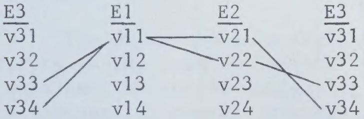

i)
E3 E1 E2 E3
v31 v11 v21 v31
v32 v12 v22 v32
v33 v13 v23 v33
v34 v14 v24 v34
v35 v15 v25 v35
v16 v26
v17
solution

Figure 3, continued.


E3 E1 E2 E3
v31 v11 v21 v31
v32 v12 v22 v32
v33 v13 v23 v33
v34 v14 v24 v34
E3 E1 E2 E3
v31 v11 v21 v31
v32 v12 v22 v32
v33 v13 v23 v33
v34 v14 v24 v34

a) No solutions

b) One solution


E3 E1 E2 E3
v31 v11 v21 v31
v32 v12 v22 v32
v33 v13 v23 v33
v34 v14 v24 v34

c) Several solutions

Figure 4. Three types of reduced state space.

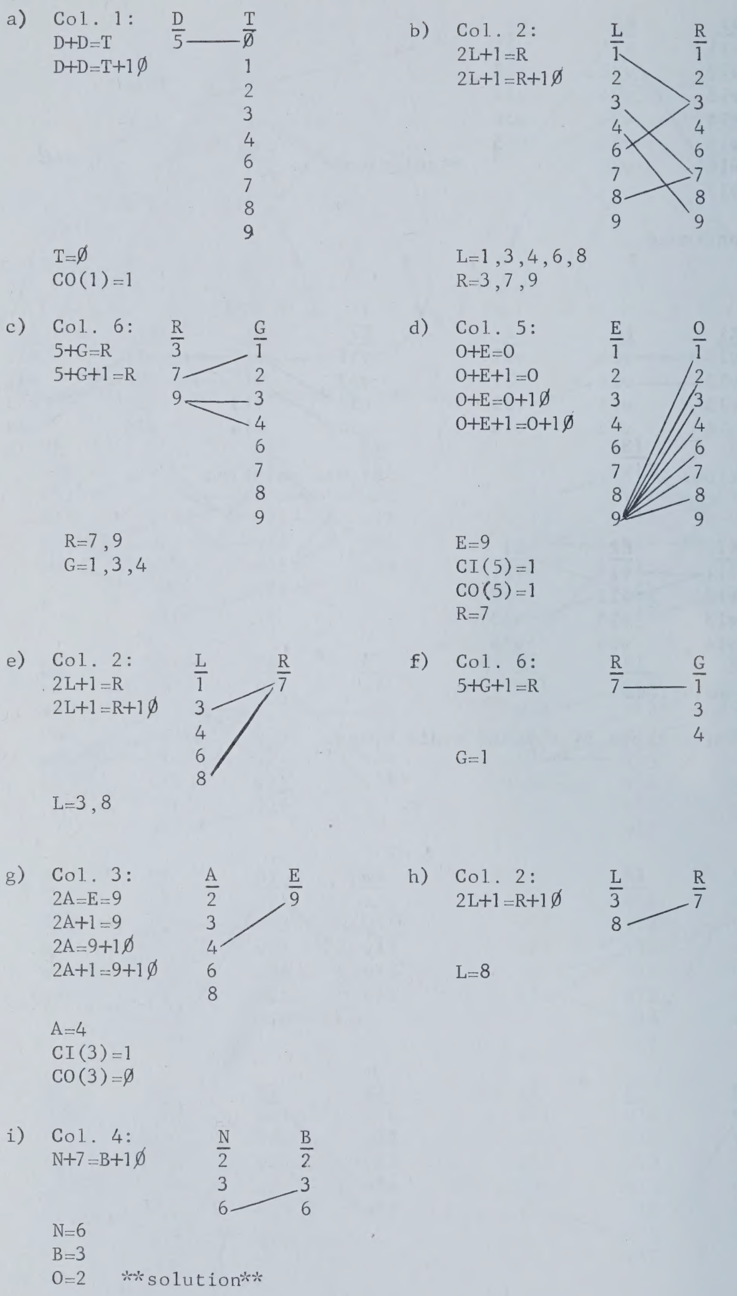

a) Col. 1:
D+D=T
D+D=T+10
D
5
T
0
1
2
3
4
6
7
8
9
T=0
CO(1)=1
b) Col. 2:
2L+1=R
2L+1=R+10
L
1
2
3
4
6
7
8
9
R
1
2
3
4
6
7
8
9
L=1,3,4,6,8
R=3,7,9
c) Col. 6:
5+G=R
5+G+1=R
R
3
7
9
G
1
2
3
4
6
7
8
9
R=7,9
G=1,3,4
d) Col. 5:
O+E=O
O+E+1=O
O+E=O+10
O+E+1=O+10
E
1
2
3
4
6
7
8
9
O
1
2
3
4
6
7
8
9
E=9
CI(5)=1
CO(5)=1
R=7
e) Col. 2:
2L+1=R
2L+1=R+10
L
1
3
4
6
8
R
7
L=3,8
f) Col. 6:
5+G+1=R
R
7
G
1
3
4
G=1
g) Col. 3:
2A=E=9
2A+1=9
2A=9+10
2A+1=9+10
A
2
3
4
6
8
E
9
A=4
CI(3)=1
CO(3)=0
h) Col. 2:
2L+1=R+10
L
3
8
R
7
L=8
i) Col. 4:
N+7=B+10
N
2
3
6
B
2
3
6
N=6
B=3
O=2 **solution**

Figure 5. Application of CS1 to DONALD+GERALD=ROBERT (given D=5). In each subfigure, the constraints used are shown on the left, and the inferences derived are shown below. CO means carry out.

NATURAL LANGUAGE BASED INFORMATION RETRIEVAL*

R. P. GABRIEL and D. L. WALTZ
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois 61801

ABSTRACT

A program which accepts natural language queries can allow a non-technical user to easily obtain information from a large non-uniform data base. To date most natural language systems have used parsers whose operation bears dubious resemblance to human language understanding and furthermore, these systems have proved to be difficult to extend to new data bases.

This paper discusses the design of a new natural language data base interrogation system which should overcome these difficulties.

INTRODUCTION

The prime obstacle for non-technical people who wish to use computers is the need to learn a special language for communicating with the machine. We are investigating ways in which computers can understand man's language. General language understanding by a computer is not possible with currently available linguistic and common-sense knowledge. However, it is conceivable that computer programs can be designed to understand natural discourse about selected subjects. This paper describes the design of a system which understands questions about a particular data base and provides answers to these questions. This research represents a significant step toward the goal of general language understanding.

OUR DATA BASE

We have obtained a data base from the Navy consisting of detailed records of aircraft maintenancy and flight information extending over a period of time. Each time a plane is serviced, a record is made including such information as the time and duration of the maintenance, who performed it, what action was taken, which parts were used or cannibalized, the manufacturers of these parts, and whether or not the service was scheduled or unscheduled. Records on the number of flights and the number of hours in the air are also kept for each plane.

A PROPOSED SCENARIO

When completed, our basic system will operate as illustrated in the following scenario. A user sits down at a terminal and introduces himself to the system; any records of previous dialogue with the user are

---

*This work is supported by the Office of Naval Research under Contract N00014-67-A-0305-0026. Auxiliary support was received from the Joint Services Electronics Program (U.S. Army, U.S. Navy, and U.S. Air Force) under Contract DAAB-07-72-C-0259.

retrieved, and the system begins accepting input (marked with an asterisk).

* We were talking about Phantoms.
Yes.
* How many Phantoms required unscheduled maintenance in April?
Thirty Phantoms.
* Of these, were any associated with radar failure?
Yes, three of them.
* What happened in these cases?
Phantom 1 required a new radar screen. Phantom 2 required replacement of a power supply. Phantom 3 crashed as a result of radar failure on a training flight.
* Did any other planes crash?
During what time period?
* April and May.
Yes, two planes crashed. A Phantom and a P3.
* Both in Appl?
Do you mean April?
* Yes.
Yes, both in April.
* Is it true that in both cases and April flights occurred while on the fuselage?
I cannot understand your question -- it is ungrammatical and meaningless.
* Is it true that im both cases the planes were fron the same squadron?
I assume you mean "Is it true that in both cases the planes were from the same squadron?" Yes, they were.
* Is there anything in common between the two planes?
Both planes were serviced on April 24th by maintenance group Hedron FMFPAC.
* Aha!

The above scenario illustrates several key issues involved in the problem of having a computer understand a question and then answer it.

1. The computer must understand what has been said. It must realize the functions of the various parts of the sentence and adequately relate them to its own understanding of the domain under discussion. Furthermore, it must be able to put all of the parts together to create a reasonable data-base search routine.

2. The computer must be able to understand what hasn't been said, but what is understandable from the context of the discussion and the current sentence. For example, the phrase "of these" can be comprehended without asking the enquirer what "these" refers to.

3. The machine must be able to know when it is failing to understand what is being said. Certainly the computer should not proceed to answer a question blindly when it is possible to report a misunderstanding. We need some way for the computer to be able to assess semantic meaningfulness while putting together a formal request.

4. Finally, the computer must be able to, upon request, provide the enquirer with sufficient information to convince him that the answer provided is indeed an answer to the question posed. (This ability is not illustrated in the dialogue above.) In general, this means that the program produced by semantic analysis must be presented to the user in an easily readable form. One solution to this problem is for the system to print out an English (or pseudo-English) translation of the program. But this means that we would need an algorithm to determine what a class of computer programs does. Not only is this a very difficult problem in our situation but it is, in its most general form, an unsolvable one.

   A more natural solution is to have an intermediate level language in which the semantic analysis program writes programs; this language is one which is easily understandable by non-technical personnel and which is reliably translatable into the lower level data base search routines.

OUTLINE OF THE SYSTEM

When a user types a question to the program, the following sequence of events occurs (refer to Figure 1):

A. The parser/semantic interpreter, using both standard and specialized vocabularies and knowledge produces a program in an intermediate level language. This interpreter maintains a set of contex registers which contain information such as the time period referred to, the last sentence and its analysis, the last answer, and so on. These registers are updated while semantic analysis is taking place. For instance, upon encountering the phrase "in April" the program modifies the time register to contain "April."

B. The program produced by the semantic interpreter is in a form gauged to be readable by people who have no technical or practical knowledge of programming languages.

For instance the question:
"Which plane, the P3 or the F4, had more occurrences of unscheduled maintenance in April?"
would be translated by the semantic analysis program into:

```
1.  Let G0001 be the set:
         (the set of all P3 such that the following are true:
              ((maintenance restrictions (unscheduled))
               (time restrictions (April))))

2.  Let G0002 be the set:
         (the set of all F4 such that the following are true:
              ((maintenance restrictions (unscheduled))
               (time restrictions (April))))

3.  If (the number of elements in G0001) is less than
         (the number of elements in G0002)
              Print "P3's had more unscheduled maintenances in April."
```

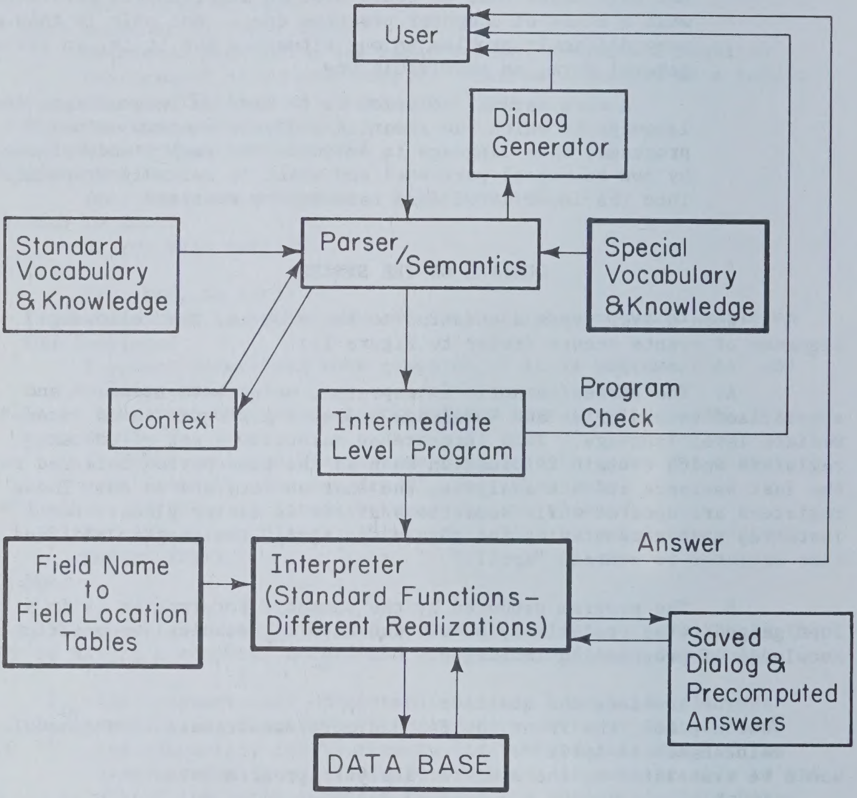

User
Dialog Generator
Standard Vocabulary & Knowledge
Parser/Semantics
Special Vocabulary & Knowledge
Context
Intermediate Level Program
Program Check
Answer
Field Name to Field Location Tables
Interpreter (Standard Functions-Different Realizations)
Saved Dialog & Precomputed Answers
DATA BASE

Figure 1

```
    Otherwise
        Print "P3's and F4's had the same number of unscheduled
        maintenances in April."
```

Whenever a set is considered by the program, a variable is generated to hold that set; thus one can naturally reference such a set with a pronoun or phrase. So in this case two sets are created, one each for the number of unscheduled maintenances performed in April on P3's and F4's. The cardinalities of the sets are compared and appropriate messages printed out.

C. This intermediate program is printed out for the user so that he can judge whether or not his request was understood. The user has the option of retyping a question if he feels the program is incorrect.

D. The intermediate program is then reliably compiled into the proper algebraic and logical combinations of primitive searching requests and executed on the data-base. During this compilation, manditory substitutions are made for such things as airplane type to airplane code and field name to field location within a record, etc.

E. Each program is stored along with intermediate results, where feasible. Thus, for instance, the previous example would store the question, the context registers, the intermediate program, and the sets G0001 and G0002; these sets are actually the sets of all records found which satisfy the search criteria, so that all other information about those maintenance actions is available in the stored results box as a precomputed answer. This storage allows the program to answer follow-up questions without another lengthy search in many cases; for instance, if the question

"Which plane, the P3 or the F4 had more occurrences of unscheduled maintenance in April?"

was followed by:

"How many of these F4's were grounded for more than two days?",

the search would be on the set G0002. However, if the first question had been:

"How many maintenances were performed?"

an unintelligent program might have saved the entire data base. The decision to store a subfile during a search depends on the estimated cost of storage versus time savings involved.

In Figure 1, the heavy edged boxes indicate those modules which need to be custom-made for each data base while the other boxes are standard. One could introduce a natural break at the intermediate level language with all modules above (including the intermediate level language) being housed in one computer, and all below being housed in another computer. The dialogue generator, shown in Figure 1, is not discussed in this paper.

NATURAL LANGUAGE

The most interesting component of the system is the linguistic analyzer. Previous approaches used have all exhibited the classical syntax/semantic dichotomy to some degree, i.e., they have all made use of two separate components to perform linguistic analysis: a syntactic component classifies the various words as parts of speech and locates them on a "parse tree"; a semantic component produces a "semantic analysis," in some form, from the parse tree and other information available. The semantic analysis yields either a set of assertions to a data-base or a program to answer a question.

This dichotomy is best exemplified by the LUNAR system written by Woods [1] to answer questions about a data base containing information on the chemical analysis of lunar rocks. First a parse tree was produced by an augmented transition network parser, and the output was subjected to semantic analysis.

In Winograd's blocks world program [2], an attempt is made to integrate the two components, but the dichotomy still exists: a well-defined grammar is used while a semantic component checks for semantic consistency as the parse proceeds.

A more semantically oriented system was developed by Schank [3]. His system utilizes a formal parser, but heavily relies on the "conceptual framework" underlying the sentence to complete the analysis and modify the parse.

We feel that this dichotomous situation is to a large degree unnecessary. In many cases, syntactic categories are best assigned to certain words after a semantic analysis has been performed.

We have observed that sentences (particularly questions) are organized in sentence types according to the kind of information they contain. The main problem of a language understanding program is to recognize the sentence type and extract the type of information it can then expect. As auxiliaries to these sentence types are certain phrase types which can occur at almost any point in almost any sentence. Usually these are propositional phrases which provide such context information as time and location. Examples are:

"in April"
"in the U.S."
"at the time"
"last year."

Semantically the auxiliary phrases fill the context registers mentioned earlier (see Figure 1).

One fault of previous parsers is that they try to treat this type of phenomenon as part of the general case. Since these auxiliary phrases can appear at many places in a sentence, they should be handled on an interrupt basis; that is, whenever it is suspected that such a phrase is present, the current processing should be halted and a special service routine entered. Part of the information contained in each phrase type is a specification of where, within each sentence type, that phrase type can appear.

Certain sentence types can have their own peculiar interrupt phrases. For example, questions that begin with "which" can have a phrase that lists the possibilities to choose from. Thus, we can have: 1) "Which plane, the F4 or the P3, had more unscheduled maintenance last month?", or 2) "Which plane had more unscheduled maintenance last month, the F4 or the P3?"

This phenomenon is unique to wh-questions, i.e., those that begin with which, what, who, when and where. Therefore, this knowledge is included as part of the specific information of wh-sentence types. Some other sentence types in our system are:

How many ...
How much ...
How did ...
Please tell me ...
Can you tell me ...
Count ...
When did ...
Can you find ...

Note that these sentence types are all indexed by the initial phrases, so that the system only needs to check the first words of a question in order to determine the question type in most cases.

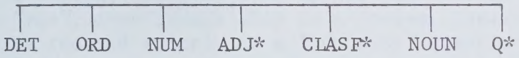
In addition to sentence types are various phrase types:

A. The prepositional phrase. Each type corresponds to a preposition and the kind of information provided by the phrase. As an example, there are at least three prepositional phrase types beginning with "in":

in ⟨physical location - NP⟩
in ⟨condition - NP⟩
in ⟨time period - NP⟩

Here the symbol "NP" means that the syntactic category of the phrase following "in" is a noun phrase.

B. The noun phrase. We have adopted Winograd's analysis of the noun phrase with some modification. In general, the form of a NP is:



where * means that more than one element of that type can occur in a noun phrase.

DET - a determiner such as the, a, etc.
ORD - a ordinal. first, last, next.
NUM - a number. one, three, at least 5
ADJ - an adjective. big, slow, expensive,...
CLASF - a classifier is a word that is a noun but can be used as an adjective. So, for example, "aircraft fuselage" has "aircraft" as a classifier.
NOUN - a noun.
Q - a qualifier, which can be a prepositional phrase or a relative clause. Examples: planes which require no maintenance the landing gear on an S-3.

Noun phrases are used to represent a class of objects - where "objects" are loosely defined - and a set of restrictions on these objects. In the case of our data-base the noun phrase will typically specify some set of records and the information of interest on these cards.

C. The verb phrase. Verb phrases are generated by the verb that occurs in them and in this regard we will be most closely following Schank. Each verb specifies the kind of conceptual structure it has, including what kinds of agents, actors, recipients, and direct objects can occur, as well as their relative position within the various sentence types.

For example, the verb "require" will have roughly the following type of definition:

time specification (interrupt)
thing that requires - require(s) - thing required

or

time specification (interrupt)
thing required - was required by - thing that requires

IMPLEMENTATION OF THE STRUCTURAL TYPES

Since most of the sentence and phrase types are signaled by specific words, the definitions of these structures are stored in memory as word definitions. Each word has as its definition a list of condition-action pairs. For phrase and sentence types, the condition is usually a pattern to be looked for in the remainder of the sentence, and the action can include code fragments, suggestions for filling in the code fragment, and possible register manipulations. The condition pattern contains suggestions about where other phrase types can occur and their meaning within the phrase.

For instance, the definition of "how" is organized as follows:

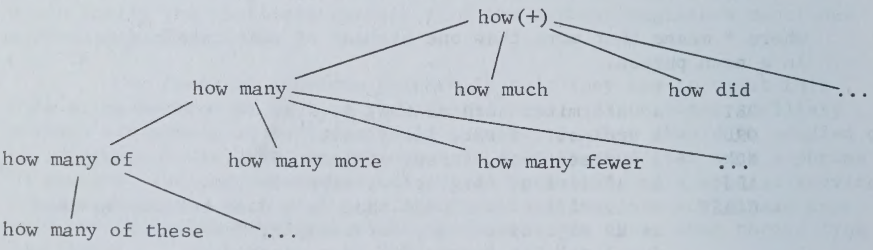


Figure 2

After encountering "how" as the first word of a question (after a possible interrupt phrase such as "in April and May" the system checks to see what the next word is - "many", "much" or others. If it is none of the ones shown on the second level of the tree, the question is unrecognizable since "how" is marked with a "+" meaning one of the daughter cases must be applicable.

Suppose the question is "how many --" and the word following "many" is neither "fewer", "none", "less", or "of". Then the definition of "how many" looks roughly like this:

```
how many
    [NP countitem (context how-many) count restrictions]
    [VP action (context how-many) action restrictions]

    ==>

    (count the number of ⟨countitem⟩ which satisfy the following
    restrictions:
        ((⟨count restrictions⟩ NP)
         (⟨action restrictions⟩ VP)
         (⟨time restrictions⟩ context-register))).
```

The system then expects a noun phrase containing a noun that tells what to count with restrictions derived from the adjectives and classifiers. Similarly, the verb phrase provides further restrictions on the count item, but the analysis of the verb phrase depends on the conceptual structure of the verb. In general there are several conceptualizations for a verb depending on the sentence type in which it is found. The sentence type is passed to the NP and VP components as the second element of the (context -) pair. This is followed by the program skeleton which contains instructions about how to fill its slots in doing the rest of the analysis. Note that some restrictions come directly from the manditory analysis of the NP and VP, whereas others come from the interrupt driven context registers. Finally, entire phrases can have a separate definition, particularly idioms. In our system, under the definition of "can", we find the definition of, "can you please tell me." Therefore, we do not require frequently encountered phrases to be parsed every time they are encountered; this precompilation of idioms also frees the system from having to understand the parsing of a phrase not really meant to be parsed. For instance, a user would not really want a program to answer the question, "can you tell me how many planes crashed in April?", with "yes", even though this is a correct answer. In our system, this question is treated exactly as a "how many" question since the idiom, "can you tell me", is understood immediately as a whole.

THE LINGUISTIC COMPONENT-EXTENDABILITY

The extendability of the linguistic component follows from the fact that the syntactic and semantic structures are stored as a part of the word definitions and not, as in other systems, as a separate entity which must be altered when a new structure is desired. Thus, in this system a new word definition tells the system what structure the word has and which structures it can enter into, thus increasing the structural flexibility of the system without distressing side effects.

CONCLUSION

Our system, though at an early stage of development, displays several unique features, both in terms of an automatic programming system for data-base retrieval and in terms of a natural language system.

The entire system has been designed specifically for ease of extendability and usage. Further, it attempts to overcome some of the faults that earlier natural language systems have exhibited. In particular, the semantic analysis is totally integrated with syntactic analysis; in fact, our view has been that syntactic analysis is a natural part of the semantic analysis and not a separate function of a system.

We hope and expect that this system will be able to mature into a general approach to natural language understanding.

REFERENCES

1. Woods, W. A., Kaplan, R. M., Nash-Webber, R., "The Lunar Science Natural Language Information System: Final Report," BBN Report No. 2378, Bolt, Beranek and Newman, Inc., Cambridge, Mass., June 1972.

2. Winograd, T., Understanding Natural Language, Academic Press, New York, 1972.

3. Schank, R., "Identification of Conceptualizations Underlying Natural Language," in Computer Models of Thought and Language, R. C. Schank and K. Colby (eds.), W. H. Freeman and Company, San Francisco, 1973.

VARIABLE-VALUED LOGIC: RECENT DEVELOPMENTS

R. S. MICHALSKI
University of Illinois
Urbana, Illinois 61801

ABSTRACT

In a variable-valued logic system (VLS), a well-formed formula and each variable in a formula can accept values from different domains which have own problem-dependent properties. Research on the development of VLSs has been carried on for several years[1-12] and has demonstrated a great attractiveness of such systems to pattern recognition, machine learning, inductive inference, unification of various combinatorial problems, etc.

The major recent developments in this area include:

- Improvements and experiments with the computer program[7] AQVAL/1 implementing inductive processes in terms of VL_1 system. This program is, at the present time, the most general and advanced program in existence of this kind (an inductive learning program) (as compared to similar known programs in the U.S.A. and U.S.S.R.).
- Work toward a new system called VL_2 (which can be characterized as a 'variable-valued first order predicate calculus').
- Development of the program OMNIBUS for:
  (i) formating and automatic quantization of the raw data for program AQVAL/1
  (ii) evaluating VL_1 formulas (deductive processes)
  (iii) gathering statistics about performance of VL_1 formulas used as classification rules.

REFERENCES

1. Michalski, R. S., "On the Quasi-Minimal Solution of the General Covering Problem," Proceedings of the V International Symposium on Information Processing (FCIP 69), Vol. A3 (Switching Circuits), Yugoslavia, Bled, October 8-11, 1969 (in English).

2. Michalski, R. S. and B. H. McCormick, "Interval Generalization of Switching Theory," Proceedings of the Third Annual Houston Conference on Computer and System Science, Houston, Texas, April 26-27, 1971.

3. Michalski, R. S. and B. H. McCormick, "Interval Generalization of Switching Theory," (an extended version of paper 2), Report No. 442, Department of Computer Science, University of Illinois, Urbana, Illinois, May 3, 1971.

4. Michalski, R. S., "A Geometrical Model for the Synthesis of Interval Covers," Report No. 461, Department of Computer Science, University of Illinois, Urbana, Illinois, June 24, 1971.

5. Michalski, R. S., "A Variable-Valued Logic System as Applied to Picture Description and Recognition," GRAPHIC LANGUAGES, Proceedings of the IFIP Working Conference on Graphic Languages, Vancouver, Canada, May 1972.

6. Michalski, R. S., "Discovering Classification Rules by the Variable-Valued Logic System VL_1," Proceedings of the Third International Joint Conference on Artificial Intelligence, Stanford, California, August 20-24, 1973.

7. Michalski, R. S., "AQVAL/1--Computer Implementation of a Variable-Valued Logic System VL_1 and Examples of Its Application to Pattern Recognition," Proceedings of the First International Joint Conference on Pattern Recognition, Washington, D.C., October 30-November 1, 1973, pp. 3-17.

8. Michalski, R. S., "VARIABLE-VALUED LOGIC: System VL_1," Proceedings of the 1974 International Symposium on Multiple-Valued Logic, West Virginia University, Morgantown, West Virginia, May 29-31, 1974.

9. Lovby, Terje, "Variable-Valued Logic Applied to Quality Control in a Clinical Laboratory," Master's Thesis, Department of Computer Science, University of Illinois, Urbana, 1974.

10. Michalski, R. S., "A Variable Decision Space Approach for Implementing a Classification System," Proceedings of the 2nd International Joint Conference on Pattern Recognition, Lyngby-Copenhagen, Denmark, August 13-15, 1974.

11. Michalski, R. S., "Learning by Inductive Inference," Proceedings of the NATO Advanced Study Institute on Computer Oriented Learning Processes, Bonas, France, August 26 - September 7, 1974.

12. Baskin, A. B., "A Comparative Discussion of Variable-valued Logic and Grammatical Inference," Report No. 663, Department of Computer Science, University of Illinois, Urbana, Illinois, July 1974.

A CONSUMER THEORY FOR DURABLE GOODS; THE AUTOMOBILE MARKET AND WELFARE IMPLICATIONS

DONALD A. HANSON
The Ohio State University
Columbus, Ohio 43210

ABSTRACT

First, a theory of the consumer is presented when he is faced with investing in fuel consuming durable goods. These results are used as part of a model of the economy which includes the automobile market. The automobile stock adjustment process is discussed as it responds to variables such as fuel price. Finally, it is proved that the stock adjustment process of a competitive market is Pareto optimum. However, important ethical issues related to intragenerational and intergenerational equity are raised.

INTRODUCTION

This paper is concerned with the welfare implications of the automobile stock adjustment process which results from an increase in fuel price. There are many problems associated with a high fuel price and automobile stock adjustment. Consumers faced with long commutes, at least until they change their travel patterns, are faced with large transportation expenses. Consumers who prefer a large automobile, perhaps those with large families, may also pay substantial fuel bills. At the same time, the market value of used large automobiles may drop and their owners then suffer a capital loss. Yet the more wealthy may continue to purchase large automobiles and hence sustain the oversupply of large used automobiles. Faced with these problems will public regulation of the automobile industry or gasoline industry increase welfare or do the economic processes at work in some sense maximize welfare?

In this paper some conditions are given which guarantee that the market equilibrium distribution of goods and services - including fuel and automobiles - is Pareto optimum. A distribution is Pareto optimum if it is feasible and if there exists no other feasible distribution such that the utility of one consumer is increased without the utility of any other consumer being decreased [1]. An exchange of goods which increases the utility of both consumers or increases the utility of one consumer and does not change the utility of the other cosnumer must be made to achieve a Pareto optimum distribution. A feasible distribution implies that supply and demand are in equilibrium.

A Pareto optimum disbribution can be obtained for any income distribution regardless of its inequities. However, from a political and ethical viewpoint the equity of the income distribution is a very important consideration. Two types of equity issues arise in the context of durable good and nonrenewable resource distribution problems. One issue is intragenerational income distribution equity. If goods are to be distributed by market prices reflecting their social value, then all consumers must be guaranteed some minimum income to participate in these markets. If fuel price were to be considerably higher than its present price, then it would seem that the standards for what is considered a minimum income should be increased. The second issue is intergenerational equity. The relative welfare of different generations depends on their inheritance of producer's capital goods, consumer's durable goods, and the amount of nonrenewable resources which they consume. In this paper a claim to wealth is assumed to be distributed among all consumers in the present gneration and to future generations as a whole.

Sufficient conditions given in this paper for a Pareto optimum distribution are that all industries are perfectly competitive, that all consumers act independently so that neither individual firms nor consumers influence market prices by their output or consumption, and further that society allocates wealth to future generations by maximizing a future social welfare function subject to a budget constraint with market prices given. This theorem is well-known for perishable goods markets [1]. However, it is not immediately clear that durable goods markets giving rise to stock adjustment processes are Pareto optimum under similiar sufficient conditions.

Clearly the real automobile industry does not satisfy this competitive market assumption. In fact it has been claimed that the firms in the automobile industry behave so as to maximize joint profits as would a monopolist [2]. A large number of papers have addressed the question of whether a monopolistic producer of a durable good would restrict its durability or quality along with restricting output [3-5]. Also the real petroleum industry is not perfectly competitive. The extent to which this lack of competition leads to inefficiency or possible public regulations or policies designed to improve efficiency are not dealt with in this paper.

CONSUMER THEORY FOR DURABLE GOODS

A consumer theory explains the behavior of a stereotype consumer in terms of his perferences, his income or wealth, and the prices facing him. His preferences are assumed to be expressed in a utility function which may include both present and future consumption. If this utility function is assumed to have a time additive form [6], then control theory or calculus of variations techniques are used to determine consumption expenditures. Aggregate consumption of perishable goods has been studied by Yaari [7]. If the consumer holds a durable good stock, then additional dynamic equations must be introduced into the theory [4]. In this paper a consumer theory is introduced in which services derived from the durable good stock require the consumption of another resource. The model is developed directly in terms of automobile stocks and fuel consumption.

Nine consumer goods are considered - an aggregate perishable consumer good c, four durable goods a_i, and four travel demand goods z_j. The durable goods a_i may be given the following interpretation: a_1 is the consumer's stock of new small automobiles, a_2 is the stock of new large automobiles, a_3 is the stock of used small automobiles, and a_4 is the stock of used large automobiles. Hence, the important assumption is made that the total automobile stock is not malleable. One cannot transform used automobiles into new automobiles or one large automobile into two small automobiles. To increase the proportion of small automobiles in the total stock requires new investment in small automobiles. If the total stock of automobiles were malleable, one could simply reduce the size of the stock by limiting gross investment. However, in order to simplify the analysis the indivisibility property of real durable goods is not assumed. That is, the consumer's automobile stock is allowed to take on continuous and fractional values. Further, maintenance, repair, and replacement of individual units of the real durable good are assumed to be captured in the depreciation of the stock. Hence, malleability is assumed within each of the four broad categories of the automobile stock. Although these assumptions are inappropriate to study behavior of any specific consumer and his durable good decisions, they seem to be a sensible representation of aggregate behavior for the economy.

In this paper travel demand will be considered to be a good for two reasons. The first is that it increases the utility derived from the aggregate perishable good c. This phenomena can be explained in two ways. According to economic theory transportation is part of the cost of a good or service [8]. It is assumed that goods and services are spatially separated and may not be deliverable to one's home location. The consumer purchases each good which he chooses to consume up to the point where its marginal utility equals its price. This price includes the price of transportation involved in consuming the good. Hence, the optimal consumption plan includes a component of travel. The size of this component depends on the price of transportation. It is possible to solve the consumer's problem in two steps.[1] The first step is to choose the optimal bundle of consumer goods subject to inequality constraints on total direct consumption expenditures c and total travel z.[2] The original utility function which depends on all the consumption goods can then be replaced with a utility function which depends only on c and z. Travel is a good with respect to the new utility function, since it has nonnegative marginal utility. Then given the prices for c and z, the optimal aggregate consumption c and total travel z are determined. The optimal travel determined in the second step implies some optimal conbination of consumer goods determined in the first step.

The second way travel can be explained to increase the utility derived from the aggregate perishable good is to think of travel as a measure of variety in the aggregate perishable good. It is assumed that one can associate with a bundle of consumer goods two characteristics - quantity and variety. Following Lancaster it is these characteristics which enter into the utility function [9].

In the specification in this paper, travel is a consumer good for another reason. It is assumed that the consumer may derive direct utility from his automobile stock and that this utility depends on his use of this stock or travel. However, a fixed coefficient linear consumption technology between automobiles, fuel and perhaps leisure time to produce services is not assumed [9]. That is, automobile stock a_j and travel z_j need not be consumed in fixed proportions. Thus both z_j and a_j are relevant independent variables to be included in the consumer's utility function.

The realistic assumption is made that the consumer can derive two types of services from his autumobile stock. The first is transportation and the second might be some combination of status, prestige and fashion. In deriving either type of service, fuel is probably an input to the consumption technology and in this sense a complementary good with the automobile stock [9]. It is clear that the traditional durable good assumption that services rendered to the consumer are proportional to the stock of the good [3-5] are either not valid in this model or the notion of service from the automobile stock must be carefully interpreted.

---

[1] These steps are analogous to the two steps involving profit maximization for the firm. The first step is to choose optimal inputs given a fixed output and obtain a cost function. The second step is to choose the optimal output.

[2] In this paper travel demand is divided into four categories. In some circumstances a fixed amount of travel with a large automobile may allow the purchase of more of a particular consumer good than with a small automobile. That is, the travel required per unit of this good is less for a large automobile. One would expect, however, that these cases are rare.

Consider further the complementary relationship between a_j and z_j. It must be assumed that there is no utility associated with z_j if a_j were zero. It would probably be assumed that if a_j were small, the marginal utility of z_j quickly would become small as z_j increases. It is also assumed in the durable good depreciation model that a_j is consumed as a consequence of travel z_j so that if a_j were small and z_j were large, a_j would soon be consumed. Also if z_j were small one would expect that the marginal utility of a_j would quickly become small as a_j increases. Finally, it is expected that for all consumers the components a_j and a_k for $j \neq k$ are substitutes and that the components of travel are also substitutes.

An example of a possible consumption pattern in conjunction with a possible market response will be given to illustrate the relationships between the goods. Suppose the price of fuel were to increase. One would expect a shift in travel demand from large to small automobiles in the short run. Due to the complementary nature of z_j and a_j, in the long run one would expect a higher proportion of small to large automobile stocks. In the medium run the automobile stocks adjust based on market prices which reflect supply and demand for large and small automobiles. But since z_j and a_j need not be consumed in fixed proportions, in the long run the ratio a_j/z_j may be higher for large automobiles since the status, prestige, fashion type of service derived from large automobiles may increase relative to the transportation service.

The mathematical model of the consumer is given as follows: Assume the consumer plans for a horizon of T years. Further assume that his utility functional is time additive which implies that the utility derived in any two periods are independent [6]. Then for the ith consumer

$$U^i = \int_0^T g^i(c^i, z^i, a^i)\, e^{-\zeta_i t}\, dt \tag{1}$$

where ζ_i is the subjective time preference parameter and z^i and a^i are vectors. It is assumed for convenience that the equations which govern the automobile stock adjustment are constant linear differential equations.

$$\begin{aligned} \dot{a}^i_j &= -\delta_j\, a^i_j - b_j\, z^i_j + \alpha^i_j \qquad j = 1,2 \\ \dot{a}^i_j &= -\delta_j\, a^i_j + \gamma_{j-2}\, a^i_{j-2} - b_j\, z^i_j + b_{j-2}\, z^i_{j-2} + \alpha^i_j \qquad j = 3,4 \end{aligned} \tag{2}$$

The b_j and δ_j are the depreciation rates associated with travel and age respectively. The α^i_j, if it is positive, represents investment by the consumers. It is assumed that the new automobile stock j = 1,2 through depreciation produces used automobiles at a rate

$$\gamma_j\, a^i_j + b_j\, z^i_j$$

where $0 < \gamma_j < \delta_j$. The initial stocks of automobiles $a^i_j(0)$ are fixed for each consumer. Also the consumer's wealth W^i is given and must be equal to the present value of consumption expenditures.

$$W^i = -\sum_{j=1}^{4} p_j(T)\, a^i_j(T) + \int_0^T \{p_K(t)\, c^i(t) + \sum_{j=1}^{4} [p_j(t)\, \alpha^i_j(t) \tag{3}$$

$$+ p_x(t)\, r_j\, z_j^i(t) - p_s(t)(\delta_j - \gamma_j)\, a_j^i(t)] + \sum_{j=3}^{4} [p_s(t)\, b_j\, z_j^i(t)]\}\ dt$$

where $\gamma_3 = \gamma_4 = 0$, credit is given for the automobile stock at T, travel with an automobile of type j consumes $r_j z_j^i$ fuel with price p_x, automobiles which "evaporate" due to depreciation are assumed to be sold for scrap at a price p_s, and all prices are discounted or forward prices [1]. Clearly $p_j \geq p_s$ for $j = 3,4$ since the price of used automobiles cannot drop below the scrap value. The consumer is assumed to choose the functions c^i, z_j^i, α_j^i to maximize (1) subject to (2) and (3) and the decision variable constraints

$$c^i \geq 0 \qquad z_j^i \geq 0 \qquad j = 1,4 \tag{4}$$

and the state space constraint

$$a_j^i \geq 0 \qquad j = 1,4. \tag{5}$$

Consider the necessary conditions which characterize the consumer's behavior. If he chooses to consume c^i, z_j^i, or a_j^i for any j at time t, then it can be shown that he will choose them up to the point at which their marginal utilies are equal to their forward price or rental. That is

$$\eta^i g_c^i e^{-\zeta_i t} = p_K \tag{6}$$

$$\eta^i g_{z_j}^i e^{-\zeta_i t} = \begin{cases} p_x r_j + (p_j - p_{j+2})\, b_j & j = 1,2 \\ p_x r_j + (p_j - p_s)\, b_j & j = 3,4 \end{cases} \tag{7}$$

$$\eta^i g_{a_j}^i e^{-\zeta_i t} = (p_j - p_s)(\delta_j - \gamma_j) - \dot{p}_j \tag{8}$$

where η^i is a constant multiplier for each consumer. (Its inverse is frequently referred to as the marginal utility of wealth.) Although (6)-(8) have simple interpretations and are the expected results, their proof is complicated by the possibility that any of the automobile stocks a_j^i may be zero imposing a state space constraint [10]. Note that the RHS of (7) is the total cost per unit of travel and that the RHS of (8) is the rental rate of the automobile stock where $\gamma_3 = \gamma_4 = 0$. A well known decomposition of this rental rate is possible in terms of undiscounted prices $\bar{p} = pe^{\rho t}$ assuming a constant market interest rate [4]. The rental rate is the depreciation rate plus the interest rate minus the rate of capital gains

$$\bar{p}_j\, e^{-\rho t}[(1-\bar{p}_s/\bar{p}_j)(\delta_j-\gamma_j) + \rho - \dot{\bar{p}}_j/\bar{p}_j] \tag{9}$$

PARETO OPTIMALITY OF COMPETITIVE MARKET EQUILIBRIUM

Pareto optimality is based on the welfare of the consumers and not the firm. However to achieve a Pareto optimum distribution requires efficient production. In this section the production sectors of the economy will be introduced in terms of three competitive industries. Then market equilibrium conditions will be given. Finally, it will be proved that a competitive market equilibrium yeilds a Pareto optimum distribution of goods and services [1].

Petroleum Industry

Consider the k^{th} firm in a competitive petroleum industry with profit function

$$\pi_1{}^k = p_x(T)\ x^k(T) + \int_0^T p_x(t)\ R^k(t)\ dt \tag{10}$$

$$x^k = -\ R^k \qquad x^k(0) \text{ fixed} \tag{11}$$

$$R^k \geqslant 0 \qquad x^k \geqslant 0 \tag{12}$$

where x^k is the remaining stock of petroleum reserves held by the k^{th} firm and r^k is the rate of consumption of this stock. Profit maximization implies that the firm sells any amount demanded whenever $p_x(t)$ reaches some constant forward price also denoted p_x.

Automobile Industry

Consider the ℓ^{th} firm in a competitive automobile assembly industry with profit function

$$\pi_2{}^\ell = \int_0^T [p_1 v_1{}^\ell + p_2 v_2{}^\ell - p_K\ C(v_1{}^\ell, v_2{}^\ell)]\ dt \tag{13}$$

$$v_1{}^\ell \geqslant 0 \qquad v_2{}^\ell \geqslant 0 \tag{14}$$

where $v_1{}^\ell$ and $v_2{}^\ell$ are the production rates for new small and large automobiles respectively and C is the cost function for the firm. Profit maximization implies p_j equals the marginal cost with respect to v_j whenever $v_j > 0$.

Other Goods Industry

Consider the m^{th} firm in a competitive industry which produces all other goods and services for the economy.

$$\pi_3{}^m = p_K(T)\ K^m(T) + \int_0^T \{p_K(t)[F(K^m, r_p^m, s^m) - I^m(t)] - p_s(t)\ s^m(t) - p_x r_p{}^m\}\ dt \tag{15}$$

$$\dot{K}^m = -\ \delta_5\ K^m + I^m \qquad K^m(0) \text{ given} \tag{16}$$

$$s^m \geqslant 0 \qquad r_p{}^m \geqslant 0 \tag{17}$$

where I^m is gross investment in the capital stock K^m, $r_p{}^m$ is the rate of petroleum input, and s^m is the rate of scrap automobile flow. It is not assumed that the production function F is homogeneous of degree one since the labor force is fixed. Profit maximization implies that each factor used in production is hired up to the point where the value of its marginal product is equal to its price (or rental rate).

Demand from Future Generations

It is assumed that the welfare of future generations depends on the capital stocks existing for their use. Suppose the welfare function is

$$U^{N+1}(\bar{X}, \bar{K}, \bar{A}) \tag{18}$$

where $\bar{X}$ is the remaining petroleum stock, $\bar{K}$ is the accumulated producer's capital stock, and $\bar{A}$ is a vector of the automobile stocks. Suppose future generations have a claim on the wealth W^{N+1} where

$$W^{N+1} = p_X \bar{X} + p_K(T) \bar{K} + \sum_{j=1}^{4} p_j(T) \bar{A}_j \tag{19}$$

It is assumed that given prices p_X, p_K, p_j at T that future generations demand quantities $\bar{X}$, $\bar{K}$, $\bar{A}_j$ at T which maximize (18) subject to (19).

Market Equilibrium

The following supply and demand relationships determine the equilibrium market forward prices p_X, p_K, p_j, p_S. It is assumed that there are N consumers and M_j firms in industry j.

$$\sum_{k=1}^{M_1} R^k = \sum_{m=1}^{M_3} r_p{}^m + \sum_{i=1}^{N} \sum_{j=1}^{4} r_j \, z_j^i \tag{20}$$

$$\sum_{\ell=1}^{M_2} v_j{}^\ell = \sum_{i=1}^{N} \alpha_j^i \qquad j = 1,2 \tag{21}$$

$$V_j = \sum_{i=1}^{N} \alpha_j^i \qquad j = 3,4 \tag{22}$$

$$\sum_{m=1}^{M_3} s^m = \sum_{j=1}^{4} [(\delta_j - \gamma_j) \sum_{i=1}^{N} a_j^i] + \sum_{j=3}^{4} [b_j \sum_{i=1}^{N} z_j^i] - V_3 - V_4 \tag{23}$$

$$V_3 \leqslant 0 \qquad V_4 \leqslant 0 \tag{24}$$

Note that $p_j > p_s$ for j = 3,4 implies $V_3 = V_4 = 0$. If $p_j = p_s$, then V_j is determined by (22).

$$\sum_{m=1}^{M_3} F(K^m, r_p{}^m, s^m) = \sum_{i=1}^{N} c^i + \sum_{m=1}^{M_3} I^\ell + \sum_{\ell=1}^{M_2} C(v_1{}^\ell, v_2{}^\ell) \tag{25}$$

Note that the output of the industry 3 is divided among consumer's perishable goods, producer's capital goods, and automobile parts.

$$\sum_{k=1}^{M_1} X^k(T) = \bar{X} \geqslant 0 \tag{26}$$

$$\sum_{m=1}^{M_3} K^m(T) = \bar{K} \geqslant 0 \tag{27}$$

$$\sum_{i=1}^{N} a_j^i(T) = \bar{A}_j \geqslant 0 \qquad j = 1,4 \tag{28}$$

Pareto Optimality Theorem

Theorem: Let p_X, p_K, p_S, p_j for $j = 1,4$ be a set of price functions on $[0,T]$ such that if firms in competitive industry 1 maximize profits (10) subject to (11), (12); if firms in competitive industry 2 maximize (13) subject to (14); if firms in competitive industry 3 maximize (15) subject to (16), (17); if each consumer maximizes his utility (1) subject to (2)-(5); if future generations maximize their welfare function (18) subject to (19); and if the market satisfies the equilibrium conditions (20)-(28), then the distribution of goods and services E^o described by the $(9N+M_1+2M_2+3M_3+2)$ functions on $[0,T]$ c^{io}, z_j^{io}, α_j^{io}, R^{ko}, $v_1^{\ell o}$, $v^{\ell o}$, I^{mo}, s^{mo}, r_p^{mo}, $\bar{V}_3^o$, V_4^o is Pareto optimum.

Proof: Suppose, to the contrary, that there exists another distribution E^1 which is feasible in the sense that (2), (4), (5), (11), (12), (14), (16), (17), (20)-(28) are satisfied and that

$$U^{i1} \geqslant U^{io} \text{ for } i = 1, N+1 \tag{29}$$

where the inequality holds strictly for at least one consumer or the future generations with index N+1. Suppose then that for $i = \sigma$

$$U^{\sigma 1} > U^{\sigma o} \text{ which implies } W^{\sigma 1} > W^{\sigma o} \tag{30}$$

since $U^{\sigma o}$ is maximized by E^o subject to $W^{\sigma o}$ and prices p_X, p_K, p_S, p_j. Further,

$$W^{i1} \geqslant W^{io} \text{ for } i = 1, N+1 \tag{31}$$

since it is true for all i for which strict inequality holds in (29), and the reverse strict inequality cannot hold if $U^{i1} = U^{io}$ since the maximum value U^{io} cannot be achieved for $W < W^{io}$. Since strict inequality holds in (31) for at least $i = \sigma$,

$$\sum_{i=1}^{N+1} [W^{i1} - W^{io}] > 0 \tag{32}$$

Also since E^o maximizes π_1^k, $k = 1, M_1$; π_2^ℓ, $\ell = 1, M_2$; π_3^m, $m = 1, M_3$ given the prices p_X, p_K, p_j, p_S

$$\sum_{k=1}^{M_1} (\pi_1^{k1} - \pi_1^{ko}) \leqslant 0, \quad \sum_{\ell=1}^{M_2} (\pi_2^{\ell 1} - \pi_2^{\ell o}) \leqslant 0, \quad \sum_{m=1}^{M_3} (\pi_3^{m1} - \pi_3^{mo}) \leqslant 0 \tag{33}$$

Finally, substracting (33) from (32) and substituting expressions for wealth and profits yields

$$p_X[(\bar{x}^1 - \sum_{k=1}^{M_1} x^{k1}(t)) - (\bar{x}^o - \sum_{k=1}^{M_1} x^{ko}(T))] \tag{34}$$

$$+ p_K(T)[(\bar{K}^1 - \sum_{m=1}^{M_3} K^{m1}(T)) - (\bar{K}^o - \sum_{m=1}^{M_3} K^{mo}(T))]$$

$$+ \sum_{j=1}^{4} p_j(T)[(\bar{A}_j{}^1 - \sum_{i=1}^{N} a_j^{i1}(T)) - (\bar{A}_j{}^o - \sum_{i=1}^{N} a_j^{io}(T))]$$

$$+ \int_0^T \{p_K(t)[(\sum_{i=1}^{N} c^{i1} + \sum_{\ell=1}^{M_2} C(v_1{}^{\ell 1}, v_2{}^{\ell 1}) + \sum_{m=1}^{M_3} (I^{m1} - F(K^{m1}, r_p{}^{m1}, s^{m1})))$$

$$- (\sum_{i=1}^{N} c^{io} + \sum_{\ell=1}^{M_2} C(v_1{}^{\ell o}, v_2{}^{\ell o}) + \sum_{m=1}^{M_3} (I^{mo} - F(K^{mo}, r_p{}^{mo}, s^{mo})))]$$

$$+ \sum_{j=1}^{2} p_j(t)[(\sum_{i=1}^{N} \alpha_j^{i1} - \sum_{\ell=1}^{M_2} v_j{}^{\ell 1}) - (\sum_{i=1}^{N} \alpha_j^{io} - \sum_{\ell=1}^{M_2} v_j{}^{\ell o})]$$

$$+ \sum_{j=3}^{4} p_j(t)[(\sum_{i=1}^{N} \alpha_j^{i1}) - (\sum_{i=1}^{N} \alpha_j^{io})]$$

$$+ p_x[(-\sum_{k=1}^{M_1} R^{k1} + \sum_{m=1}^{M_3} r_p{}^{m1} + \sum_{j=1}^{4} \sum_{i=1}^{N} r_j\, z_j^{i1})$$

$$- (-\sum_{k=1}^{M_1} R^{ko} + \sum_{m=1}^{M_3} r_p{}^{mo} + \sum_{j=1}^{4} \sum_{i=1}^{N} r_j\, z_j{}^{io})]$$

$$+ p_s(t)[(\sum_{m=1}^{M_3} s^{m1} - \sum_{j=1}^{4} (\delta_j - \gamma_j)(\sum_{i=1}^{N} a_j^{i1}) - \sum_{j=3}^{4} \sum_{i=1}^{N} b_j\, z_j{}^{i1})$$

$$- (\sum_{m=1}^{M_3} s^{mo} - \sum_{j=1}^{4} (\delta_j - \gamma_j)(\sum_{i=1}^{N} a_j^{io}) - \sum_{j=3}^{4} \sum_{i=1}^{N} b_j\, z_j{}^{io})]\} \, dt > 0$$

However, by (20)-(28) the expression in (37) must be zero. This is easily shown for two cases $p_j = p_s$ and $p_j > p_s$ for $j = 3$ or 4. Therefore, E^O must be Pareto optimum.

CONCLUSIONS

The model presented here has several implications for the automobile stock adjustment process. The first conclusion is that even though a_j^i is a state variable and α_j^i is a decision variable, the a_j^i is determined by consumer i as if it were a decision variable. That is according to (8) at each instant a stock of a_j^i is held up to the point at which its marginal utility is proportional to its rental rate. The relationship between the marginal utility of a_j^i (MUa_j^i) and the other goods has been discussed. In particular, MUa_j^i would probably increase with z_j^i. The z_j^i from (7) depends on MUz_j^i, the price of fuel p_x, the fuel efficiency r_j, and the cost of automobile consumption per unit of travel which depends on p_j. The two price components p_x and p_j may move in the same or opposite directions. It is clear that in determining a_j^i and z_j^i that the market prices p_j are critical. These prices are determined by market equilibrium equations (21), (22). Hence, although the automobile stock may adjust to new equilibrium levels based on changes in exogenous variables such as the wealth of future generations W^{N+1}, the market always stays in (or close to) equilibrium (assuming stability) by adjusting the prices p_j. The main result of this paper is that this automobile adjustment process is Pareto optimal if the production sectors are competitive and prices are freely determined.

Concluding on an ethical point, if fuel price p_x rises due to a political or ethical decision to increase W^{N+1}, then the poor will suffer a disproportionate hardship in their relative reduction in wealth. Thus, it seems appropriate to compensate the poor, in addition to future generations, leading to a more equitable income distribution.

REFERENCES

[1] E. Malinvaud, Lectures on Microeconomic Theory, (Amsterdam: North Holland, 1972), Chapter 4.

[2] L. J. White, The Autumobile Industry since 1945, (Cambridge: Harvard University Press, 1971).

[3] P. L. Swan, "Durability of Consumption Goods," American Economic Review, Vol. 60 (1970), pp. 884-894.

[4] R. W. Parks, "The Demand and Supply of Durable Goods and Durability," American Economic Review, Vol. 64 (1974), pp. 37-55.

[5] P. L. Swan, "Optimum Durability, Second-Hand Markets, and Planned Obsolescence," Journal of Political Economy, Vol. 80 (1972), pp. 575-585.

[6] G. Hadley and M. C. Kemp, Variational Methods in Economics, (Amsterdam: North-Holland, 1971), pp. 364-66.

[7] M. E. Yaari, "On the Consumer's Lifetime Allocation Process," International Economic Review, Vol. 5 (1964), pp. 304-317.

[8] H. O. Nourse, Regional Economics, (New York: McGraw-Hill, 1968), Chapter 2.

[9] K. J. Lancaster, "A New Approach to Consumer Theory," Journal of Political Economy, Vol. 74 (1966), pp. 132-157.

[10] A. E. Bryson and Y. C. Ho, Applied Optimal Control, (Waltham, Mass.: Blaisdell Publishing Compnay, 1969), pp. 117-118.

A DYNAMIC MODEL OF EXCHANGE RATE IN RELATION TO INTERNATIONAL TRADING

K. G. SUN, Graduate Assistant, and O. D. OSBORNE, Assistant Professor
Electrical and Computer Engineering Department
Oregon State University, Corvallis, Oregon 97331

ABSTRACT

It is popular belief that the volatility of the exchange rate is due to the balance of payments, which in turn is mainly influenced by the balance of trade. A closed-loop dynamic economic model is proposed to describe and analyze the effect of the international trade of a country on the relative stability of its foreign exchange rate. The relative stability of the foreign exchange rate is seen to be moderated by a country's inventory. The inventory level is related to the supply (exports) and demand (imports), with the close monitoring of the monetary authority. The monetary authority, seeking to maintain a desired inventory level, would set the desired exchange rate. This behavior is at the same time influenced by the desire for relative stability of the exchange rate. The inventory concept is an adaption of the model of Fitch and Frick[1] [1].

INTRODUCTION

After decades of relative neglect, economic theory, especially in its mathematical form, has taken new life since the advance of computers. The renaissance has included many aspects of international economics. Although the interrelationship between exchange rate and international trade via balance of trade, is one of the oldest themes in economics, until very recently there have been very few well developed dynamic (as opposed to comparative-static) models in the literature. One of the main purposes of this study, which is described in its title and delimited by its size, is to consider some aspects of the impact of the pattern of international trading in terms of a very simple but roughly "unifying" framework. It is an outline of the theory of the rate of exchange, concise in form and elementary in approach. The model attempts to conceptualize the relationship between international trade and exchange rate, and the possible role of the central authority in this relationship. The model is particularly concerned with examining the interconnections between a few variables that seem to be strategic. In this initial effort, in order to be able to concentrate the analysis on the dominant forces (exports and imports) that characterize international trading, frictions and disturbances (such as tariffs, speculation effects, capital flows, and transportation cost, etc.) are disregarded.

THE DYNAMIC MODEL

A macro-economic model of international trading is formulated explicitly in the area of exports and imports. The model is continuous and dynamic in nature. It incorporates the inventory concept of the central authority. A time moving model is used to analyze these questions because economic growth alters the underlying relation between the trade and exchange rate as time progresses. The dynamic continuous model is illustrated in Figure 1.

The Model

The change of the inventory level of a country depends on the deficit or surplus of its external trade. When these changes accumulate over a period of time, they become the country's accumulated inventory. This process expressed in the mathematical form would be the following equation:

---

[1] Sincere appreciation is extended to Dr. Frick and Dr. Fitch for their generous, stimulating discussions and interests in this study.

$\overset{\circ}{\underline{\Phi}}$ $= \underline{EX}(t) - \underline{IM}(t)$

where $EX(t)$ = exports of goods in dollars

$IM(t)$ = imports of goods in dollars (1)

$\overset{\circ}{\underline{\Phi}}(t)$ = change in inventory level

In order to adopt the most intelligent combination of budgetary and monetary policy conceivable, the country's monetary authority would set a desired inventory level to control its external trade. It is conceived that the difference between the market inventory level and the central authority desired inventory level would change the value of the currency, consequently the price level. Furthermore, the price level would support itself under certain circumstances, i.e., in peace and prosperity. This established the equation:

$\overset{\circ}{R}(t)$ $= \alpha \underline{R}(t) - \beta[\underline{\Phi}(t) - \underline{\Phi}_d(t)]$

$R(t)$ = exchange rate

$\overset{\circ}{\underline{R}}(t)$ = change in exchange rate (2)

$\underline{\Phi}_d(t)$ = desired inventory level

$\underline{\Phi}(t)$ = inventory level

α = exchange rate adjustment coefficient

β = inventory adjustment coefficient

The inventory level $\underline{\Phi}(t)$, corresponds to the classical inventory notion, strives to maintain a desired inventory level $\underline{\Phi}_d(t)$, rather than to fluctuate freely.

Exports and Imports

In the econometric analysis of international trade, the emphasis has traditionally been on income and price. The reader is referred to [2] and [3]. Due to the theory of purchasing power parity, [4] and [5], under the international trade environment in the time of peace and prosperity, as being the primary determinant of the exchange rate, the export and import equations are formulated as follows:

$\log \underline{IM}$ $= A_o + A_1 \log \underline{Y} + A_2 \log (\underline{R}/R(o)) + U$

$\log \underline{EX}$ $= B_o + B_1 \log \underline{YW} + B_2 \log (\underline{R}/R(o)) + V$

A_o, B_o = constants

A_1 = income elasticity for the country

B_1 = income elasticity for the rest of the world

A_2 = exchange rate elasticity for the country's imports

B_2 = exchange rate elasticities for the country's exports

U, V = error terms

$\underline{R}$ = current exchange rate

$R(o)$ = par exchange rate

$\underline{Y}$ = yearly income of the country

$\underline{YW}$ = yearly income of the rest of the world

$\underline{IM}$ = imports

$\underline{EX}$ = exports

Throughout the paper, the logarithmic equations are used because of their generally superior fit and ease of interpretation. The least-square fitting is used to generate the elasticities in Equation (3).

Inventory

From the Equation (1) the difference between the exports $\underline{EX}(t)$ and imports $\underline{IM}(t)$ is the change in inventory $\overset{\circ}{\underline{\Phi}}(t)$. The change in inventory is smoothed and accumulated by applying Simpson's 3/8 rule. The result is designated as that of the market inventory level $\underline{\Phi}(t)$

$$\underline{\Phi}(t) = \int_{t_o}^{t_o + nh} \overset{\circ}{\underline{\Phi}}(t) + \Phi(o) \qquad (4)$$

where $\Phi(o) = 0.0$

$t = t_o + nh \qquad n = 1, 2, 3 \ldots$

h = unit intertal year
month

Simpson's 3/8 rule is applied to average the small changes on the curves as a reflection of ignoring the minor disturbances in reality. The desired inventory level $\underline{\Phi}_d(t)$ is designated as the average of the last twelve market inventory levels.

$$\underline{\Phi}_d(t) = \sum_{n = t-12}^{t} \underline{\Phi}(n) \Big/ 12 \qquad (5)$$

The number twelve is selected as it designates total months of a fiscal year.

The difference between the market inventory level and the desired inventory level indicates the variance of the estimation of the central authority and the actual market effect. This in turn causes a change in the rate of exchange rate by the parameter $-\beta$.

Exchange Rate

In the period of peace and prosperity, the exchange rate would tend to support itself regardless of the short duration of the actual trade position. Rewriting Equation (2), in state space form

$$\overset{\circ}{\underline{R}}(t) = [-\alpha \ \ -\beta] \begin{bmatrix} \underline{R}(t) \\ \underline{\Phi}(t) - \underline{\Phi}_d(t) \end{bmatrix} \qquad (6)$$

Again Simpson's 3/8 rule is applied to smooth and accumulate the changes of exchange rate so that the future exchange rate is

$$\underline{R}(t+1) = \int_{t_o}^{t_o + nh} \overset{\circ}{\underline{R}}(t)dt + R(o) \qquad (7)$$

where $R(o)$ = the exchange rate of the base year

$t = t_o + nh$ $n = 1, 2, 3...$

h = time interval year

month

The future exchange rate then is in turn affecting the future exports and imports.

SIMULATION

Japan's domestic economy, international trading, and exchange rate for the period (1960-1970) are used to simulate the model. In the simulation, the open loop system is first investigated. This is executed by inserting the actual data of Japan's monthly exports and imports[2] into the model to generate the exchange rate. The closed feedback network of the model is completed by using two different versions of Equation (3) to generate Japan's exports and imports instead of their actual data. The variables in Equation (3) are first expressed in index form by dividing each variable respectively by its corresponding value for the base year. This index version of export and import equations incorporated into the model is Simulation A. For Simulation B, the export and import values in Equation (3) are each inflated respectively by corresponding export and import prices while the other variables are expressed as indexes. The monthly income is derived by dividing the term for the year income in Equation (3) and not the yearly income[3] itself by twelve. The monthly indices for export and import prices are repetitions of their quarterly indices[2].

CONCLUSIONS

Despite the lack of reliable data, the computed monthly exports and imports of Simulation A and Simulation B reported in yearly form are comparable to the actual data. The exchange rate of Simulation A has a closer fit to the open loop simulation, up until 1968. From 1968 to 1970, exchange rate from Simulation B is by far superior. There is no upper or lower limit imposed on the exchange rates and indeed some of the worst case simulations that have been experienced do have unacceptable and uncontrollable exchange rates.

Although the authors do not delude themselves that such a model gives a convincing picture of the real world, the introduction of inventory concept for linking the international trade to the exchange rate is believed to be a novel and powerful idea [1]. The simulations show the attribution. The model projects, in essence, a tool, a dynamic method.

REFERENCES

[1] Fitch, J.B. and P.A. Frick. A Dynamic Macro-Model for an Open Regional Economy. Milwaukee Symposium on Automatic Control. March, 1974. pp. 280-285.

[2] Houthakker, H.S. and S.P. Magee. Income and Price Elasticities in World Trade. Review of Economics and Statictics 51(2): 111-125. 1969.

---

[2] Data source: United Nation Monthly Bulletin of Statistics (various issues).

[3] Data source: United Nation Statistical Yearbook (various volumes).

[3] Ujiie, J. and P. Yeung. Japan's Economic Growth and Balance of Trade. The American Economist. 16(2): 106-122. 1972.

[4] Yeager, L.B. International Monetary Relations. New York, Harper and Row, 1966. pp. 504.

[5] Gailliot, H.J. Purchasing Power Parity as an Explanation of Long-Term Changes in Exchange Rates. J. Money, Credit, Banking. 2(3):348-357. 1970.

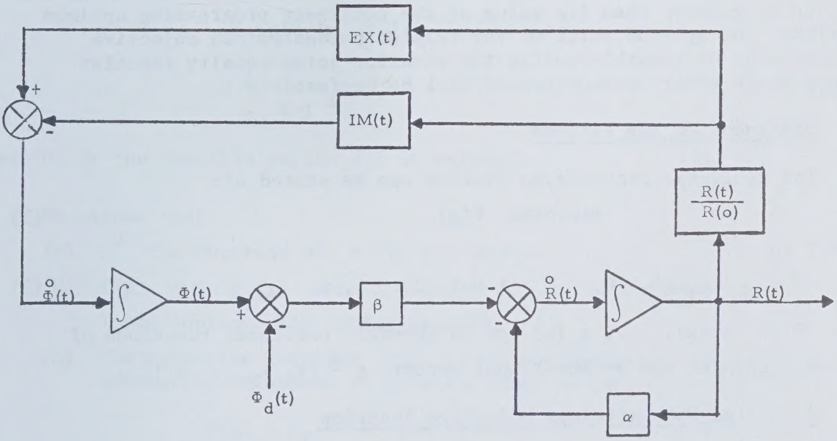

EX(t)
+
-
IM(t)
R(t)/R(o)
$\overset{o}{\Phi}(t)$
Φ(t)
+
-
β
-
-
$\overset{o}{R}(t)$
R(t)
$\Phi_d(t)$
α

Figure 1. Block diagram for the model.

A NEGATIVE-POSITIVE BARRIER METHOD FOR NONLINEAR PROGRAMMING

CHRISTAKIS CHARLAMBOUS

Department of Combinatorics and Optimization

University of Waterloo, Waterloo, Ontario, Canada

Abstract. A new approach to nonlinear programming is presented. The original nonlinear programming problem is formulated as a sequence of unconstrained optimization problems. It is shown that this sequence will converge to the nonlinear programming optimum. The proposed negative-positive barrier function has certain advantages over the usual barrier and benalty function methods. In addition, by minor modification of the new unconstrained objective function it is possible from a single optimization to see if there exists a feasible region.

1. Introduction

A large number of methods are available to solve the nonlinear programming problem including many variations of the barrier function methods due to Fiacco and McCormick [1,2], the Rosen's gradient projection method [3,4] and the recent work by Powell, Hesteness and Rockfellar [5,6,7].

The proposed unconstrained objective function is defined simultaneously inside and outside the feasible region, unlike the barrier function methods which are undefined outside the feasible region. If we start from a point which is not feasible and the value of the constrained function is greater than its value at the nonlinear programming optimum point then the optimum point of the related unconstrained objective function will be feasible unlike the exterior point penalty function methods where their optimum points will be nonfeasible.

2. Statement of the Problem

The nonlinear programming problem can be stated as:

$$\text{minimize} \quad F(\underset{\sim}{x}) \tag{1}$$

subject to

$$g_i(\underset{\sim}{x}) \geq 0, \qquad i = 1, 2, \ldots, m \tag{2}$$

where $F(\underset{\sim}{x})$, $g_1(\underset{\sim}{x})$, ..., $g_m(\underset{\sim}{x})$ are in general, nonlinear functions of the components of the n-dimentional vector $\underset{\sim}{x} \overset{\Delta}{=} [x_1 \; x_2 \; \ldots \; x_n]^T$.

3. A Related Unconstrained Objective Function

Let us consider the following unconstrained objective function.

This work was supported by the National Research Council of Canada through a Postdoctorate Fellowship to the author and by the Department of Combinatorics and Optimization of the University of Waterloo. The author is greatly indebted to Dr. J.W. Bandler of McMaster University and to Dr. A. Conn of the University of Waterloo for useful discussions.

$$U(\underset{\sim}{x}, r, F^j) = M(\underset{\sim}{x}) \left(\sum_{i=0}^{m} \left(\frac{w_i f_i(\underset{\sim}{x})}{M(\underset{\sim}{x})} \right)^q \right)^{\frac{1}{q}} \quad \text{for} \quad M(\underset{\sim}{x}) \neq 0 \tag{3}$$

$$= 0 \quad \text{for} \quad M(\underset{\sim}{x}) = 0$$

where

$$f_0(\underset{\sim}{x}) = (F(\underset{\sim}{x}) - F^j) \quad , \quad F^j \text{ is a scalar} \tag{4(a)}$$

whose importance will become apparent later.

$$f_i(\underset{\sim}{x}) = -rg_i(\underset{\sim}{x}) \quad , \; r > 0 \; , \; i = 1, 2, \ldots, m \tag{4(b)}$$

$$M(\underset{\sim}{x}) = \max_{0 \le i \le m} f_i(\underset{\sim}{x}) \tag{5}$$

$$q = p \text{ x sign } (M(\underset{\sim}{x})) \; , \; p \text{ is a constant } > 1 \tag{6}$$

$$\text{for } M(\underset{\sim}{x}) > 0 \begin{cases} w_i = 1 \text{ for } f_i(\underset{\sim}{x}) > 0 \; , \; i = 0, 1, \ldots, m \\ w_i = 0 \text{ for } f_i(\underset{\sim}{x}) < 0, \; i = 0, 1, \ldots, m \end{cases} \tag{7}$$

and for $M(\underset{\sim}{x}) < 0$, $w_i = 1$, $i = 0, 1, \ldots, m$.

4. Definitions and Assumptions.

Let

$$D_i \triangleq \{\underset{\sim}{x} | f_i(\underset{\sim}{x}) \le 0\} \; , \; i = 0, 1, \ldots, m \tag{8a}$$

$$D \triangleq \bigcap_{i=1}^{m} D_i \tag{8b}$$

then D is the feasible region (if it exists).

We assume that

(a) D^0 the interior of D is not empty.

(b) $F(\underset{\sim}{x})$ and $g_i(\underset{\sim}{x})$ for $i = 1, 2, \ldots, m$ belong to class $C^{(1)}$ (once continuously differentiable).

(c) The objective function $F(\underset{\sim}{x})$ is bounded below on D, i.e. we assume the existence of greatest lower bound L_f such that

$$F(\underset{\sim}{x}) \ge L_f > -\infty \quad \text{for} \quad \underset{\sim}{x} \in D \; .$$

(d) The set $S \triangleq \{\underset{\sim}{x} | F(\underset{\sim}{x}) \le c, \; x \in D\}$ is bounded below for every finite value of c.

(e) $F(\underset{\sim}{x})$ and $-g_i(\underset{\sim}{x})$ for $i = 1, 2, \ldots, m$ are convex.

$\overset{\vee}{\underset{\sim}{x}}$ will denote the optimum point of the problem (1), (2).

Under the stated assumptions the unconstrained objective function (3) is continuous with continuous first partial derivatives, except when $M(\underset{\sim}{x}) = 0$ and two or more of the $f_i(\underset{\sim}{x})$ for $i = 0, 1, \ldots, m$ are equal to zero in which case $U(\underset{\sim}{x}, r, F^j)$ is continuous but the gradient is discontinuous.

3.1 Illustrative Example [8]

Consider the following simple example:

minimize

$$F(\underset{\sim}{x}) = (x_1 - 2)^2 + (x_2 - 1)^2$$

Subject to

$$g_1(\underset{\sim}{x}) = x_2 - x_1^2 \geq 0$$

$$g_2(\underset{\sim}{x}) = -x_1 - x_2 + 2 \geq 0$$

The optimum point occurs at the point $\overset{\vee}{\underset{\sim}{x}} = [1 \quad 1]^T$ and $F(\overset{\vee}{\underset{\sim}{x}}) = 1$.

Figures 1, 2 and 3 show contours of the unconstrained objective function $U(\underset{\sim}{x}, r, F^j)$ for $p = 2$, $r = 1$, and for different values of F^j.

For Figure 1, $F^j = 2.5$ ($> F(\overset{\vee}{\underset{\sim}{x}})$) and it can be seen that the unconstrained objective function defined by equation (3) has the following property:

$$U(\underset{\sim}{x},r,F^j)\begin{cases} < 0 & \text{for } \underset{\sim}{x} \text{ lying in the interior of the region } D_0 \cap D \\ = 0 & \text{for } \underset{\sim}{x} \text{ lying on the boundary of the region } D_0 \cap D \\ > 0 & \text{for } \underset{\sim}{x} \text{ lying outside the region } D_0 \cap D \end{cases} \qquad (9)$$

Therefore we can start from any point feasible or nonfeasible to minimize $U(\underset{\sim}{x}, r, F^j)$. Note that, for the points $\underset{\sim}{x}$ where one or more of the constraints are equal to zero and $\underset{\sim}{x} \in D_0 \cap D$, $U(\underset{\sim}{x}, r, F^j) = 0$, while in the most usual barrier function methods the barrier function is equal to infinity. As in the barrier function methods, is very unlikely in the linear search to hit the constraint boundary, it is very unlikely in our case to hit the points of discontinuous derivatives of $U(\underset{\sim}{x}, r, F^j)$ (sharp points on the contour $U(\underset{\sim}{x}, r, F^j) = 0$). Due to this fact it is possible to use efficient gradient methods to minize $U(\underset{\sim}{x}, r, F^j)$. Note the $F(\overset{\vee}{\underset{\sim}{x}}) < F(\overset{\vee}{\underset{\sim}{x}}^j) < F^j$, where $\overset{\vee}{\underset{\sim}{x}}^j$ is the optimum point of U.

For Figure 2, $F^j = 0.5$ ($< F(\overset{\vee}{\underset{\sim}{x}})$). Since $D_0 \cap D = \emptyset$, $U(\underset{\sim}{x}, r, F^j) > 0$ for all $\underset{\sim}{x}$'s We do not have any discontinuous derivatives and also $F^j < F(\overset{\vee}{\underset{\sim}{x}}^j) < F(\underset{\sim}{x})$. For Figure 3, $F^j = 1$ ($= F(\overset{\vee}{\underset{\sim}{x}})$) and it can be seen that $\overset{\vee}{\underset{\sim}{x}}^j = \overset{\vee}{\underset{\sim}{x}}$.

4. The Algorithms

4.1 Algorithm 1

For algorithm 1 we set $F^j = F(\underset{\sim}{x}^0)$, where $\underset{\sim}{x}^0$ is the starting point and we minimize the unconstrained objection function given by

Equation (3) for strictly increasing sequence of $r(r_k, k = 1, 2, \ldots)$ such that $\lim_{k\to\infty} r_k \to \infty$. The value of p is kept constant throughout the algorithm.

It is important to note that the point $\underset{\sim}{x}^0$ can be anywhere. If $F(\underset{\sim}{x}^0) > L_f$ (note that $\underset{\sim}{x}^0$ might not be feasible) then the points $\check{\underset{\sim}{x}}^k$ $(k = 1, 2, \ldots)$ will be inside the feasible region, $F(\check{\underset{\sim}{x}}^k) < F(\underset{\sim}{x}^0)$ and at the point $\check{\underset{\sim}{x}}^k$, $q = -p$. ($\check{\underset{\sim}{x}}^k$ is the optimum point of (3) with $r = r_k$). If $F(\underset{\sim}{x}^0) < L_f$ then the points $\check{\underset{\sim}{x}}^k$ $(k = 1, 2, \ldots)$ will be outside the feasible region, $F(\underset{\sim}{x}^0) < F(\check{\underset{\sim}{x}}^k) < L_f$ (from Theorem 2) and $q = p$. If $F(\underset{\sim}{x}^0) = F(\check{\underset{\sim}{x}})$ then $\check{\underset{\sim}{x}}^1 = \check{\underset{\sim}{x}}$.

4.2 Theorem 1. If $F(\underset{\sim}{x})$ is convex and $g_i(\underset{\sim}{x})$ for $i = 1, 2, \ldots, m$ are concave, then $U(\underset{\sim}{x}, r, F^j)$ is convex.

Proof. See [9].

4.3 Theorem 2. If $F(\underset{\sim}{x}^0) < F(\check{\underset{\sim}{x}})$, then by minimizing (3) we will get

$$F^j < F(\check{\underset{\sim}{x}}^k) < F(\check{\underset{\sim}{x}})$$

Proof. For $\underset{\sim}{x} \in D_0$, $U(\underset{\sim}{x}, r, F^j)$ becomes

$$V(\underset{\sim}{x}) = M(\underset{\sim}{x})\left(\sum_{i=1}^{m}\left(\frac{w_i f_i(\underset{\sim}{x})}{M(\underset{\sim}{x})}\right)^q\right)^{\frac{1}{q}} \qquad (9)$$

(Note that $f_0(\underset{\sim}{x})$ is neglected). The optimum point of (9) will be inside the feasible region (if the feasible region exists). Let us suppose that we minimize (9) subject to the constraint $f_0(\underset{\sim}{x}) \le 0$. In this case the optimum point $\check{\underset{\sim}{x}}^c$ will lie on the boundary of D_0. Now, consider (3) at the point $\check{\underset{\sim}{x}}^c$. Since $p > 1$, $\underset{\sim}{\nabla} U(\check{\underset{\sim}{x}}^c) = \underset{\sim}{\nabla} V(\check{\underset{\sim}{x}}^c) \neq \underset{\sim}{0}$ and therefore $\check{\underset{\sim}{x}}^k \notin D_0$, where $\underset{\sim}{\nabla} \overset{\Delta}{=} \left[\frac{\partial}{\partial x_1} \; \frac{\partial}{\partial x_2} \; \ldots \; \frac{\partial}{\partial x_n}\right]^T$. In other words $F(\check{\underset{\sim}{x}}^k) > F^j$.

On the other hand if we assume that $F(\check{\underset{\sim}{x}}^k) \ge F(\check{\underset{\sim}{x}})$, then $U(\check{\underset{\sim}{x}}^k) > U(\check{\underset{\sim}{x}})$ (for $\underset{\sim}{x}^k \neq \check{\underset{\sim}{x}}$) and therefore it contradicts the assumption. (Note that at the point $\check{\underset{\sim}{x}}$, $\nabla U(\check{\underset{\sim}{x}}) = \underset{\sim}{\nabla} F(\check{\underset{\sim}{x}}) \neq \underset{\sim}{0}$). Therefore $F(\check{\underset{\sim}{x}}^k) < F^j$.

It is important to note that, if at the optimum point of (9) $M \le 0$, then, there exist a feasible region, and if $M > 0$ then, feasible region does not exist. That is, with only one optimization we will know if there exists a feasible region, unlike Fiacco and McCormick approach where we have to do a sequence of optimizations [1]. From now on we will often denote $U(\underset{\sim}{x}, r, F^j)$ by $U(\underset{\sim}{x}, r)$.

4.4 Lemma 1. If $r_{k+1} > r_k$ and $F(\underset{\sim}{x}^0) \neq L_f$, then,

$$|U(\check{\underset{\sim}{x}}^{k+1}, r_{k+1})| > |U(\check{\underset{\sim}{x}}^k, r_k)|$$

Proof. See [9] .

4.5 Theorem 3. Under the stated conditions

$$\lim_{k\to\infty} U(\overset{\vee}{\underset{\sim}{x}}{}^{k}, r_k) \to L_f - F(\underset{\sim}{x}^{0})$$

$$\lim_{k\to\infty} \overset{\vee}{\underset{\sim}{x}}{}^{k} = \overset{\vee}{\underset{\sim}{x}} .$$

Proof. See [9] .

4.6 Lemma 2.

$$\text{If} \begin{cases} F(\underset{\sim}{x}^{0})>L_f \text{ then } F(\overset{\vee}{\underset{\sim}{x}}{}^{k+1})<F(\overset{\vee}{\underset{\sim}{x}}{}^{k}), \ \sum_{i=1}^{m} (g_i(\overset{\vee}{\underset{\sim}{x}}{}^{k+1}))^{-p} > \sum_{i=1}^{m} (g_i(\overset{\vee}{\underset{\sim}{x}}{}^{k}))^{-p} \\ F(\underset{\sim}{x}^{0})<L_f \text{ then } F(\overset{\vee}{\underset{\sim}{x}}{}^{k+1})>F(\overset{\vee}{\underset{\sim}{x}}{}^{k}), \ \sum_{i=1}^{m} w_i(-g_i(\overset{\vee}{\underset{\sim}{x}}{}^{k+1}))^{p} < \sum_{i=1}^{m} w_i(-g_i(\overset{\vee}{\underset{\sim}{x}}{}^{k}))^{p} \end{cases}$$

Proof. See [9] .

4.7 Algorithm 2.

Another possible algorithm is to keep r constant and set $F^j = F(\overset{\vee}{\underset{\sim}{x}}{}^{k})$, where $\overset{\vee}{\underset{\sim}{x}}{}^{k}$ is the optimum point of U with the value of F^j equal to $F(\overset{\vee}{\underset{\sim}{x}}{}^{k-1})$ $(F(\overset{\vee}{\underset{\sim}{x}}{}^{0}) \equiv F(\underset{\sim}{x}^{0}))$. Although this algorithm is self controlled it was found that it was too slow.

Another possibility is to change r as in algorithm 1 and F^j as mentioned above. This is called algorithm 2. The proof of convergence is similar to that of algorithm 1 and is omitted.

5. Examples.

Two examples are going to be considered so that to illustrate the usefulness of this approach to nonlinear programming. Fletcher's unconstrained minimization algorithm [10] is going to be used to minimize the unconstrained objective function U. One function evaluation includes evaluation of all first derivatives. For both examples p = 2.

6.1 The Rosen-Suzuki Problem.

For the Rosen-Suzuki problem

$$F(\underset{\sim}{x}) = x_1^2 + x_2^2 + 2x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4$$

Subject to

$$-x_1^2 - x_2^2 - x_3^2 - x_4^2 - x_1 + x_2 - x_3 + x_4 + 8 \ge 0$$

$$-x_1^2 - 2x_2^2 - x_3^2 - 2x_4^2 + x_1 + x_4 + 10 \ge 0$$

$$-2x_1^2 - x_2^2 - x_3^2 - 2x_1 + x_2 + x_4 + 5 \ge 0$$

The solution is

$$F = -44,\ x_1 = 0,\ x_2 = 1,\ x_3 = 2,\ x_4 = -1$$

Table 1 shows the progress of both Algorithms from a feasible starting point and from a non-feasible starting point. The code used is: A1F - Algorithm 1 using the feasible starting point, A1NF - Algorithm 2 using the non-feasible starting point, A2F - Algorithm 2 using the feasible starting point, and A2NF - Algorithm 2 using the non-feasible starting point.

6.2 Wong Problem [11]

In this case,

$$\begin{aligned} F(\underset{\sim}{x}) = {} & (x_1 - 1)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 - 11)^2 \\ & + 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7 \end{aligned}$$

subject to

$$\begin{aligned} - 2x_1^2 - 3x_2^4 - x_3 - 4x_2^2 - 5x_5 + 127 &\ge 0 \\ - 7x_1 - 3x_2 - 10x_3^2 - x_4 + x_5 + 282 &\ge 0 \\ -23x_1 - x_2^2 - 6x_6^2 + 8x_7 + 196 &\ge 0 \\ - 4x_1^2 - x_2^2 + 3x_1x_2 - 2x_3^2 - 5x_6 + 11x_7 &\ge 0 \end{aligned}$$

The solution is

$$\begin{aligned} & F = 680.63,\ x_1 = 2.3305,\ x_2 = 1.9514, \\ & x_3 = -0.47754,\ x_4 = 4.3657,\ x_5 = -0.62449, \\ & x_6 = 1.0381,\ x_7 = 1.5942. \end{aligned}$$

Table 2 shows the progress of both algorithms from a feasible starting point and a nonfeasible starting point respectively.

7. A Transformed Objective Function

The above algorithms have the disadvantage that the controlling parameter r has to tend to infinity and this produces ill conditioned problems. This difficulty can be overcome by transforming the objective function U given by Eq. (3), into the following unconstrained objective function.

$$U_t(\underset{\sim}{x}, \underset{\sim}{r}, F^j, \underset{\sim}{t}) = M(\underset{\sim}{x}) \left[\left(\frac{w_0(F(\underset{\sim}{x}) - F^j)}{M(\underset{\sim}{x})} \right)^q + \sum_{i=1}^{m} \left(\frac{w_i(-r_i g_i(\underset{\sim}{x}) + t_i)}{M(\underset{\sim}{x})} \right)^q \right]^{1/q} \qquad (10)$$

where $\underset{\sim}{r} = [r_1\ r_2\ \dots\ r_m]^T$, $\underset{\sim}{t} = [t_1\ t_2,\ \dots\ t_m]^T$

$$M(\underset{\sim}{x}) = \max_{1 \le i \le m} \{F(\underset{\sim}{x}) - F^j,\ -r_i g_i(\underset{\sim}{x}) + t_i\}$$

$$\text{for } M(\underset{\sim}{x}) \geq 0 \begin{cases} w_i=1, \text{ for } -r_i g_i(\underset{\sim}{x})+t_i>0, \ i=1,2,\ldots,m, \ w_0=1, \text{ if } F(\underset{\sim}{x})-F^j>0 \\ w_i=0, \text{ for } -r_i g_i(\underset{\sim}{x})+t_i<0, \ i=1,2,\ldots,m, \ w_0=0, \text{ if } F(\underset{\sim}{x})-F^j<0 \end{cases}$$

and for $M(\underset{\sim}{x}) < 0$, $w_i = 1$, $i = 0, 1, \ldots, m$.

7.1 Theorem 4 If the Kuhn-Tucker conditions for optimality of the nonlinear programming problem are satisfied at $\overset{\vee}{\underset{\sim}{x}}$ then if $t_i = F(\overset{\vee}{\underset{\sim}{x}}) - F^j$ and $r_i = u_i$, $1 \leq i \leq m$ then the point $\overset{\vee}{\underset{\sim}{x}}$ will be an optimum point of $U_t(\underset{\sim}{x}, \underset{\sim}{r}, F^j, \underset{\sim}{t})$, where u_i, $1 \leq i \leq m$ are the Kuhn-Tucker multipliers.

Space does not permit a proof of this theorem.

Although, in practice, we do not know the values of the Kuhn-Tucker multipliers and the value of $F(\overset{\vee}{\underset{\sim}{x}})$, we can get very good estimates for their values if we are close to the solution.

At each optimum point of U_t we have

$$\underset{\sim}{\nabla} F(\underset{\sim}{x}) - \sum_{i=1}^{m} u_i(\underset{\sim}{x}) \, \underset{\sim}{\nabla} \, g_i(\underset{\sim}{x}) = \underset{\sim}{0}$$

$$\Delta F(\underset{\sim}{x}) = \sum_{i=1}^{m} u_i(\underset{\sim}{x}) \, g_i(\underset{\sim}{x})$$

where

$$u_i(\underset{\sim}{x}) = \left(w_i(-r_i g_i(\underset{\sim}{x})+t_i)/M(\underset{\sim}{x}) \right)^{q-1} \Big/ \left((F(\underset{\sim}{x})-F^j)/M(\underset{\sim}{x}) \right)^{q-1}$$

If $\underset{\sim}{x}$ is close to the optimum, then $u_i(\underset{\sim}{x})$ will give good estimate for u_i and $\Delta F(\underset{\sim}{x})$ will give good estimate for $F(\underset{\sim}{x})-F(\overset{\vee}{\underset{\sim}{x}})$. This suggests the following algorithm.

1. Set $k=0$, $t_i=0$, $r_i=1$, $i=1,2,\ldots,m$, $F^j=F(\underset{\sim}{x}^{\circ})$
2. Minimize U_t
3. If at the optimum point reached
   $$\max_{1 \leq i \leq m} \left| u_i(\overset{\vee}{\underset{\sim}{x}}^{k}) g_i(\overset{\vee}{\underset{\sim}{x}}^{k}) \right| > 0.1 \quad \text{go to } 5$$
4. Set $k=k+1$, $r_i=u_i(\overset{\vee}{\underset{\sim}{x}}^{k})$, $t_i=-\Delta F(\overset{\vee}{\underset{\sim}{x}}^{k})$, $i=1,2,\ldots,m$, $F^j=F(\overset{\vee}{\underset{\sim}{x}}^{k})$ and go to 2.
5. Set $k=k+1$, $r_i=10r_i$, $i=1,2,\ldots,m$, and go to 2.

Using this algorithm on the two previous problems we got the results shown in Table 3. The code used is: FSP - Feasible Starting Point, NFSP - Nonfeasible Starting Point.

TABLE 3. Results Using the Transformed Objective function

k	$F(\underset{\sim}{\check{x}}^{k})$				Function Evaluations per Iteration			
	Rosen-Suzuki Problem		Wong Problem		Rosen-Suzuki Problem		Wong Problem	
	FSP	NFSP	FSP	NFSP	FSP	NFSP	FSP	NFSP
1	-10.9012	-56.8532	701.95	642.05	15	14	29	30
2	-26.8313	-44.9252	689.49	679.57	17	26	22	44
3	-39.0292	-44.0115	682.97	680.62	22	47	25	72
4	-42.8780	-44.000007*	681.12	680.63005*	23	17	33	21
5	-43.7557	-44.0000000	680.74		23	23	36	
6	-43.9472		680.63006*		24		22	
7	-44.00003*				16			
8	-44.000003				23			

* denotes the first time, we used step 4 of the algorithm.

FSP: $\underset{\sim}{\check{x}}^{8} = [0. \quad 0.999999 \quad 2.000001 \quad -1.000000]^T$

NFSP: $\underset{\sim}{\check{x}}^{5} = [0. \quad 1.000000 \quad 2.000000 \quad -1.000000]^T$

FSP: $\underset{\sim}{\check{x}}^{6} = [2.3305 \quad 1.9514 \quad -0.47753 \quad 4.3657 \quad -0.62449 \quad 1.0381 \quad 1.5942]^T$

NFSP: $\underset{\sim}{\check{x}}^{4} = [2.3305 \quad 1.9514 \quad -0.47754 \quad 4.3657 \quad -0.624487 \quad 1.0381 \quad 1.5942]^T$

REFERENCES

[1] FIACCO, A.V., and McCORMICK, G.P., The Sequential Unconstrained Minimization Technique for Nonlinear Programming, A Primal-Dual Method, Management Science, Vol. 10, No. 2, 1964.

[2] FIACCO, A.V., and McCORMICK, G.P., Computational Algorithm for the Sequential Unconstrained Minimization Technique for Nonlinear Programming, Management Science, Vol. 10, No. 4, 1964.

[3] ROSEN, J.B., The Gradient Projection Method for Nonlinear Programming, Part I: Linear Constraints, J. Soc. Ind. Appl. Math., Vol. 8, No. 1, 1960.

[4] ROSEN, J.B., The Gradient Projection Method for Nonlinear Programming, Part II: Nonlinear Constraints, J. Soc. Ind. Appl. Math., Vol. 9, 1961.

[5] POWELL, M.J.B., A method for Nonlinear Constraints in Minimization Problems, Ed. R. Fletcher, Academic Press, London, 1969.

[6] HESTENES, M.R., Multiplier and Gradient Methods, Journal of Optimization Theory and Applications, Vol. 4, 1969.

[7] ROCKAFELLAR, R.T., The Multiplier Method of Hestenes and Powell Applied to Convex Programming, Journal of Optimization Theory and Applications, Vol. 12, No. 6, 1973.

[8] LASDON, L.S., Optimization Theory of Large Systems, New York, Macmillan, 1970.

[9] CHARALAMBOUS, C., A Negative-Positive Barrier Method for Nonlinear Programming, University of Waterloo, Department of Combinatorics and Optimization, Research Report 74-2, Jan. 1974.

[10] FLETCHER, A New Approach to Variable Metric Algorithms, Computer Journal, Vol. 13, No. 3, 1970.

[11] ASAADI, J., A Computational Comparison of Some Non-Linear Programs, Math. Prog. Vol. 4, 1973.

TABLE 1. Results for Rosen-Suzuki Problem

Feasible starting point: $x_1 = x_2 = x_3 = x_4 = 0$, $F(\underset{\sim}{x}^\circ) = 0$

Nonfeasible starting point: $x_1 = x_2 = 2$, $x_3 = 5$, $x_4 = 0$, $F(\underset{\sim}{x}^\circ) = -67$

k	r	$F(\underset{\sim}{x}^{vk})$				Function Evaluations per Iteration			
		A1F	A1NF	A2F	A2NF	A1F	A1NF	A2F	A2NF
1	1	-10.9012	-56.8532	-10.9012	-56.8532	15	14	15	14
2	10	-26.8313	-44.9252	-31.2595	-44.5461	17	26	18	29
3	10^2	-39.0292	-44.0115	-42.6157	-44.0003	22	47	17	94
4	10^3	-42.8780	-44.0001	-43.9651		23	86	35	
5	10^4	-43.7557		-43.9998		23		40	
6	10^5	-43.9472				24			
7	10^6	-43.9886				30			
8	10^7	-43.9975				21			
9	10^8	-43.9995				25			

A1F: $\underset{\sim}{x}^{v9} = [-0.00001 \quad 0.00000 \quad 1.99997 \quad -0.99999]^T$

A1NF: $\underset{\sim}{x}^{v4} = [\ 0. \quad 1. \quad 2. \quad -0.99999]^T$

A2F: $\underset{\sim}{x}^{v5} = [\ 0. \quad 1. \quad 1.99999 \quad -0.00000]^T$

A2NF: $\underset{\sim}{x}^{v3} = [\ 0.00002 \quad 1.00001 \quad 2.00002 \quad -0.99997]^T$

TABLE 2. Results for Wong Problem

Feasible starting point: $x_1 = x_6 = x_7 = 1$, $x_2 = 2$, $x_3 = x_5 = 0$, $x_4 = 4$, $F(\underset{\sim}{x}^\circ) = 714$

Nonfeasible starting point: $x_1 = x_2 = x_6 = 3$, $x_3 = x_7 = 0$, $x_4 = 5$, $x_5 = 1$, $F(\underset{\sim}{x}^\circ) = 605$

k	r	$F(\underset{\sim}{x}^{vk})$				Function Evaluations per Iteration			
		A1F	A1NF	A2F	A2NF	A1F	A1NF	A2F	A2NF
1	1	701.95	642.05	701.95	642.05	29	30	29	30
2	10	689.49	679.57	686.23	680.09	22	44	24	52
3	10^2	682.97	680.62	681.02		25	72	38	
4	10^3	681.12		680.64		33		60	
5	10^4	680.74				36			
6	10^5	680.65				34			

A1F: $\underset{\sim}{x}^{v6} = [2.3292. \quad 1.9514 \quad -0.47713 \quad 4.3657 \quad -0.62449 \quad 1.0379 \quad 1.5945]^T$

A1NF: $\underset{\sim}{x}^{v3} = [2.3307 \quad 1.9514 \quad -0.47747 \quad 4.3658 \quad -0.62448 \quad 1.03812 \quad 1.5942]^T$

A2F: $\underset{\sim}{x}^{v4} = [2.3302 \quad 1.9514 \quad -0.47744 \quad 4.3657 \quad -0.62449 \quad 1.0381 \quad 1.5943]^T$

A2NF: $\underset{\sim}{x}^{v2} = [2.3396 \quad 1.9528 \quad -0.47901. \quad 4.3720 \quad 0.62419 \quad 1.0394 \quad 1.5927]^T$

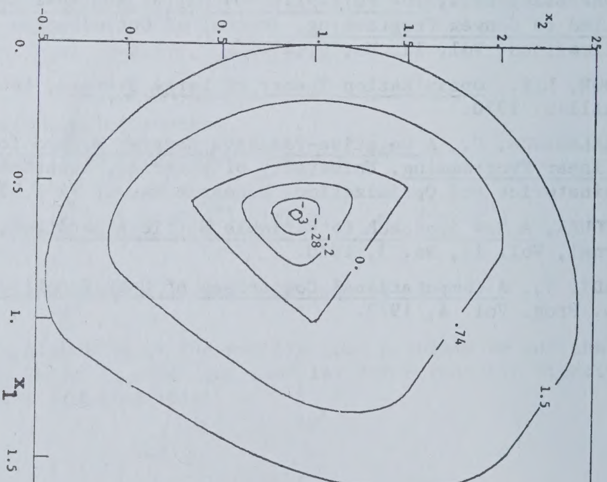
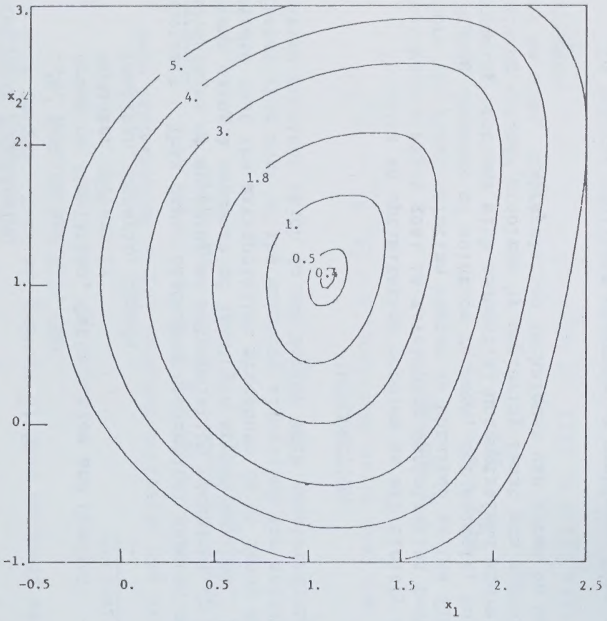
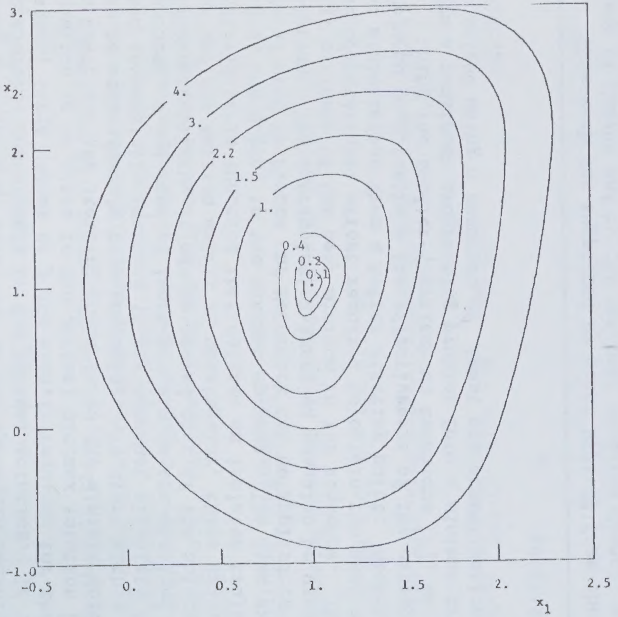


Fig. 1



3.
x_2
2.
1.
0.
-1.
-0.5
0.
0.5
1.
1.5
2.
2.5
x_1
5.
4.
3.
1.8
1.
0.5
0.3

Fig. 2



3.
x_2
2.
1.
0.
-1.0
-0.5
0.
0.5
1.
1.5
2.
2.5
x_1
4.
3.
2.2
1.5
1.
0.4
0.2
0.1

Fig. 3

COMPUTATIONAL MERITS OF EXTRAPOLATION IN LEAST pTH APPROXIMATION AND NONLINEAR PROGRAMMING

J.W. BANDLER and W.Y. CHU
Group on Simulation, Optimization and Control
McMaster University
Hamilton, Ontario, Canada

Abstract This paper discusses theoretical aspects and some computational advantages of applying an extrapolation technique in solving minimax problems using a sequence of least pth approximations. Numerical results indicate that the extrapolation procedure is at least as efficient as or faster than most of the existing algorithms for solving minimax approximation problems which do not employ such acceleration techniques.

INTRODUCTION

To solve an optimization problem as efficiently as possible has always been a major goal in developing optimization techniques. In problems where the final solution vector is obtained as the final entry of a converging sequence of solution vectors, for example, solving nonlinear programming problems with sequential unconstrained minimization techniques, solving minimax problems by employing least pth approximation, the process may be slow. Extrapolation techniques can often be used to speed up convergence.

In solving nonlinear programming problems, Fiacco and McCormick [1] - [2] and Lootsma [3] have shown that, under certain assumptions, the problem variables, on the trajectory of minima of the sequence of unconstrained functions, can be developed as functions of the parameter r. This provides a theoretical basis for an extrapolation technique that significantly accelerates convergence to the optimal solution. In least pth approximation, large values of p are usually required for the least pth optimal solution to be close to the optimal minimax solution [4]-[6]. Depending on how close the starting point is to the minimax solution, the process may be unnecessarily time-consuming. To start with a small value of p and then sequentially increase it may somewhat alleviate this problem. In this approach, a sequence of least pth minima are obtained. Under conditions of differentiability, it can be proved that if the objective function is convex and its Hessian matrix is nonsingular, there exists a continuously differentiable function that defines an isolated trajectory of least pth minima converging to the minimax optimum. The same extrapolation technique as used by Fiacco and McCormick can be applied to estimate the minimax solution. The extrapolation function can also be used to predict an initial estimate of the next minimum in the sequence. The effect is to reduce the computational effort required to obtain the next minimum by beginning the minimization from a better starting point. Several numerical examples are used to illustrate the effectiveness of the extrapolation technique.

Using the Bandler-Charalambous technique [7], we can readily transform a nonlinear programming problem into a minimax problem that may then be solved using a sequence of least pth approximations together with extrapolation.

This work was supported by grant 9931-39 from the Defence Research Board of Canada and by the National Research Council of Canada through grant E3556 and a Scholarship to W.Y. Chu.

BASIC FORMULAS

A brief review of the formulas used in solving the test examples will be presented.

Generalized Least pth Objective

The generalized least pth objective function [8] to be minimized with respect to $\boldsymbol{\phi}$ is

$$U(\boldsymbol{\phi},p) = \begin{cases} M(\boldsymbol{\phi}) \left[\sum\limits_{i\varepsilon K} \left(\dfrac{e_i(\boldsymbol{\phi})}{M(\boldsymbol{\phi})} \right)^q \right]^{\frac{1}{q}} & \text{for } M(\boldsymbol{\phi}) \neq 0 \\ 0 & \text{for } M(\boldsymbol{\phi}) = 0 \end{cases} \tag{1}$$

where $e_i(\boldsymbol{\phi})$ is a set of m+1 real error functions, $\boldsymbol{\phi} \overset{\Delta}{=} [\phi_1 \; \phi_2 \ldots \phi_n]^T$

$$q \overset{\Delta}{=} p \text{ sgn } M(\boldsymbol{\phi}) \tag{2}$$

$$M(\boldsymbol{\phi}) \overset{\Delta}{=} \max_{i\varepsilon I} \; e_i(\boldsymbol{\phi}) \tag{3}$$

$$K = \begin{cases} J \overset{\Delta}{=} \{i \mid e_i(\boldsymbol{\phi}) > 0, \; i\varepsilon I\} & \text{if } M(\boldsymbol{\phi}) > 0 \\ I \overset{\Delta}{=} \{1,2,\ldots, m+1\} & \text{if } M(\boldsymbol{\phi}) < 0 \end{cases} \tag{4}$$

The gradient vector of the objective function is given by

$$\nabla U(\boldsymbol{\phi},p) = \left[\sum_{i\varepsilon K} \left(\frac{e_i(\boldsymbol{\phi})}{M(\boldsymbol{\phi})} \right)^q \right]^{\frac{1}{q} - 1} \sum_{i\varepsilon K} \left(\frac{e_i(\boldsymbol{\phi})}{M(\boldsymbol{\phi})} \right)^{q-1} \nabla e_i(\boldsymbol{\phi}) \quad \text{for } M(\boldsymbol{\phi}) \neq 0 \tag{5}$$

From (1) and (5) we note that if the $e_i(\boldsymbol{\phi})$ are continuous with continuous first partial derivatives then, under the stated conditions, the objective function is continuous everywhere with continuous first partial derivatives (except possibly when $M(\boldsymbol{\phi}) = 0$ and two or more maxima are equal). Efficient gradient methods, e.g., Fletcher's method [9], can be employed to perform the minimization.

Minimax Approach to Nonlinear Programming

The nonlinear programming problem of minimizing $f(\boldsymbol{\phi})$ subject to

$$g_i(\boldsymbol{\phi}) \geq 0, \; i = 1,2,\ldots, m, \tag{6}$$

can be transformed into the following unconstrained objective (see Bandler and Charalambous [7])

$$V(\boldsymbol{\phi},\alpha) = \max_{1 \leq i \leq m} [f(\boldsymbol{\phi}), \; f(\boldsymbol{\phi}) - \alpha g_i(\boldsymbol{\phi})] \tag{7}$$

where α is positive satisfying

$$\frac{1}{\alpha} \sum_{i=1}^{m} u_i < 1 \tag{8}$$

where the u_i's are the Kuhn-Tucker multipliers at the optimum. The minimization of $V(\boldsymbol{\phi},\alpha)$ with respect to $\boldsymbol{\phi}$ is a minimax problem and may be solved by minimizing the generalized least pth objective with

$$e_i(\boldsymbol{\phi}) \triangleq f(\boldsymbol{\phi}) - \alpha g_i(\boldsymbol{\phi}) \quad , \quad i = 1,2,\ldots,m \tag{9}$$

$$e_{m+1}(\boldsymbol{\phi}) \triangleq f(\boldsymbol{\phi}) \tag{10}$$

using a very large value of p or a sequence of p values with extrapolation.

Extrapolation Polynomials [2]

Suppose the generalized least pth objective function $U(\boldsymbol{\phi},p)$ has been uniquely minimized for $1 < p_1 < \ldots < p_k < \infty$ at $\boldsymbol{\phi}\left(\frac{1}{p_1}\right),\ldots,\boldsymbol{\phi}\left(\frac{1}{p_k}\right)$.

Let $p' \triangleq \frac{1}{p}$. A polynomial in p' that yields $\boldsymbol{\phi}(p_1'),\ldots,\boldsymbol{\phi}(p_k')$ is given by

$$\boldsymbol{\phi}(p_i') = \sum_{j=0}^{k-1} \boldsymbol{a}_j (p_i')^j \quad , \quad i = 1,\ldots,k, \tag{11}$$

where the $\boldsymbol{a}_j$ are n-component vectors. The determinant of the matrix of coefficients is the Vandermonde determinant and is nonzero if $p_i' \neq p_j'$ for $i \neq j$ in which case we have a unique solution for the $\boldsymbol{a}_j$. Then $\sum_{j=0}^{k-1} \boldsymbol{a}_j (p')^j$ is an approximation of $\boldsymbol{\phi}(p')$ on $[0,p_1']$, and $\boldsymbol{\phi}(0) = \check{\boldsymbol{\phi}}$ (the minimax solution) is approximated by $\boldsymbol{a}_0$.

Now, the exact Taylor series expansion of $\boldsymbol{\phi}(p_i')$ in p_i' about $\boldsymbol{\phi}(0)$ is

$$\boldsymbol{\phi}(p_i') = \sum_{j=0}^{k-1} (p_i')^j \frac{D^j\boldsymbol{\phi}(0)}{j!} + \boldsymbol{\varepsilon}^i \quad , \quad i = 1,\ldots,k, \tag{12}$$

where

$$D\boldsymbol{\phi}(p') \triangleq \left[\frac{d\phi_1(p')}{dp'} \cdots \frac{d\phi_n(p')}{dp'}\right]^T \tag{13}$$

and $\boldsymbol{\varepsilon}^i$ is an error term. It can be shown that the difference between $\boldsymbol{a}_0$ and $\boldsymbol{\phi}(0)$ is of the order of $(p_1')^k$. Thus, as $p_1' \to 0$, $\boldsymbol{a}_0 \to \boldsymbol{\phi}(0)$. In addition, the estimates using k minima are better than those using k-1 minima. With $p_{i+1}' = p_i'/c$ $(c > 1)$, the particular structure of these equations renders the use of an extrapolation procedure according to the Richardson-Romberg principle [10] to estimate $\boldsymbol{a}_0$.

If $\boldsymbol{\phi}_j^i$, $i = 1,\ldots,k$, $j = 1,\ldots, i-1$ signifies the jth order estimate of $\boldsymbol{\phi}(0)$ after i minima have been obtained, with p_1' being the initial value of p', then we have

$$\boldsymbol{\phi}_0^i = \boldsymbol{\phi}\left(\frac{p_1'}{c^{i-1}}\right) \quad , \quad i = 1,\ldots,k \tag{14}$$

and

$$\boldsymbol{\phi}_j^i = \frac{c^j \boldsymbol{\phi}_{j-1}^i - \boldsymbol{\phi}_{j-1}^{i-1}}{c^j - 1} \quad , \quad \begin{array}{l} i = 2,\ldots,k \\ j = 1,\ldots,i-1 . \end{array} \tag{15}$$

The "best" estimate of $\boldsymbol{\phi}(0)$, namely $\boldsymbol{a}_0$, is given by

$$\boldsymbol{\phi}(0) \simeq \boldsymbol{\phi}^{k}_{k-1} = \boldsymbol{a}_0 \quad . \tag{16}$$

The extrapolation formula (15) can also be used to estimate the next minimum of the objective function $U(\boldsymbol{\phi},p)$, e.g., the (k+1) th minimum. Setting $i = k+1$ in (15) and solving for $\boldsymbol{\phi}^{k+1}_{j-1}$, we have

$$\boldsymbol{\phi}^{k+1}_{j-1} = \frac{(c^{j}-1)\boldsymbol{\phi}^{k+1}_{j} + \boldsymbol{\phi}^{k}_{j-1}}{c^{j}} \tag{17}$$

Noting that $\boldsymbol{a}_0 = \boldsymbol{\phi}^{k}_{k-1} = \boldsymbol{\phi}^{k+1}_{k-1}$ from (16) and using the values previously obtained from (15), we can evaluate (17) for $j = k-1, k-2, \ldots, 1$. The last computation will give the required estimate $\boldsymbol{\phi}^{k+1}_{0}$. This estimate can be used as the starting point for the (k+1)th minimization of $U(\boldsymbol{\phi},p)$. As more minima are achieved, the estimate eventually improves. This accelerates the entire process by substantially reducing the effort required to minimize the successive U functions.

THEORETICAL JUSTIFICATION

We require an isolated trajectory of least pth minima which is a continuously differentiable function in p' for $1 > p' > 0$ and therefore can be expanded as a Taylor series about $p' = 0$. To justify this we assume

(i) The error functions $e_i(\boldsymbol{\phi})$ for $i \varepsilon I$ are convex and have continuous (k+1) th order ($k \geq 1$) partial derivatives with respect to $\boldsymbol{\phi}$.

(ii) The Hessian matrix of U is nonsingular for any real $\boldsymbol{\phi}$ and for every $1 > p' > 0$.

At the minimizing point $\boldsymbol{\phi}(p')$ we have

$$\boldsymbol{\nabla} U(\boldsymbol{\phi}(p'),p) = \left\{ \sum_{i \varepsilon K} \left[\frac{e_i(\boldsymbol{\phi}(p'))}{M(\boldsymbol{\phi}(p'))} \right]^{q} \right\}^{\frac{1}{q} - 1} \sum_{i \varepsilon K} \left[\frac{e_i(\boldsymbol{\phi}(p'))}{M(\boldsymbol{\phi}(p'))} \right]^{q-1} \boldsymbol{\nabla} e_i(\boldsymbol{\phi}(p')) = \boldsymbol{0} \ . \tag{18}$$

Since by assumption the Hessian matrix of U is nonsingular, the implicit function theorem assures us that $\boldsymbol{\phi}(p')$ is a continuously differentiable vector function of p' for $1 > p' > 0$. In other words, we have an isolated trajectory of unconstrained local minima of U.

It is possible to be explicit about the derivatives of $\boldsymbol{\phi}(p')$ with respect to p' for $1 > p' > 0$. For convenience, let

$$e_{ip'} \triangleq e_i(\boldsymbol{\phi}(p')) \tag{19}$$

$$M_{p'} \triangleq M(\boldsymbol{\phi}(p')) \quad . \tag{20}$$

Since (18) is an identity in $\frac{1}{q}$ (or rather p'), we can differentiate with respect to p', obtaining

$$\begin{aligned}\left[\sum_{i\in K}\left(\frac{e_{ip'}}{M_{p'}}\right)^{q}\right]^{\frac{1}{q}-1}\sum_{i\in K}\Bigg\{&\left(\frac{e_{ip'}}{M_{p'}}\right)^{q-1}\underset{\sim}{\nabla}(\underset{\sim}{\nabla}e_{ip'})^{T}\,D\underset{\sim}{\phi}(p')\\ &+(q-1)\left(\frac{e_{ip'}}{M_{p'}}\right)^{q-2}\left(\frac{\underset{\sim}{\nabla}e_{ip'}}{M_{p'}}\right)(\underset{\sim}{\nabla}e_{ip'})^{T}\,D\underset{\sim}{\phi}(p')\\ &-(\operatorname{sgn} M_{p'})\,q^{2}\left(\frac{e_{ip'}}{M_{p'}}\right)^{q-1}\ln\left(\frac{e_{ip'}}{M_{p'}}\right)\underset{\sim}{\nabla}e_{ip'}\Bigg\}=\underset{\sim}{0}\end{aligned}\tag{21}$$

Now

$$\begin{aligned}\underset{\sim}{\nabla}(\underset{\sim}{\nabla}U(\underset{\sim}{\phi}(p'),p))^{T}=\left[\sum_{i\in K}\left(\frac{e_{ip'}}{M_{p'}}\right)^{q}\right]^{\frac{1}{q}-1}\sum_{i\in K}\Bigg\{&\left(\frac{e_{ip'}}{M_{p'}}\right)^{q-1}\underset{\sim}{\nabla}(\underset{\sim}{\nabla}e_{ip'})^{T}\\ &+(q-1)\left(\frac{e_{ip'}}{M_{p'}}\right)^{q-2}\left(\frac{\underset{\sim}{\nabla}e_{ip'}}{M_{p'}}\right)(\underset{\sim}{\nabla}e_{ip'})^{T}\Bigg\}\end{aligned}\tag{22}$$

Equation (21) can hence be written as

$$\begin{aligned}\underset{\sim}{\nabla}(\underset{\sim}{\nabla}U(\underset{\sim}{\phi}(p'),p))^{T}\,D\underset{\sim}{\phi}(p')-(\operatorname{sgn} M_{p'})q^{2}&\left[\sum_{i\in K}\left(\frac{e_{ip'}}{M_{p'}}\right)^{q}\right]^{\frac{1}{q}-1}\\ &\cdot\sum_{i\in K}\left\{\left(\frac{e_{ip'}}{M_{p'}}\right)^{q-1}\ln\left(\frac{e_{ip'}}{M_{p'}}\right)\underset{\sim}{\nabla}e_{ip'}\right\}=\underset{\sim}{0}\end{aligned}\tag{23}$$

where $D\underset{\sim}{\phi}(p')$ was defined in (13). Hence,

$$\begin{aligned}D\underset{\sim}{\phi}(p')=(\operatorname{sgn} M_{p'})\left\{\underset{\sim}{\nabla}(\underset{\sim}{\nabla}U(\underset{\sim}{\phi}(p'),p))^{T}\right\}^{-1}&q^{2}\left[\sum_{i\in K}\left(\frac{e_{ip'}}{M_{p'}}\right)^{q}\right]^{\frac{1}{q}-1}\\ &\cdot\sum_{i\in K}\left\{\left(\frac{e_{ip'}}{M_{p'}}\right)^{q-1}\ln\left(\frac{e_{ip'}}{M_{p'}}\right)\underset{\sim}{\nabla}e_{ip'}\right\}\end{aligned}\tag{24}$$

If we differentiate (23) with respect to p', we shall find that the existence of the same inverse is required for $D^{2}\underset{\sim}{\phi}(p')$ to exist as required for $D\underset{\sim}{\phi}(p')$. In addition, $D^{2}\underset{\sim}{\phi}(p')$ requires the existence of the third partial derivatives of $e_{i}(\underset{\sim}{\phi})$ with respect to $\underset{\sim}{\phi}$. By continuing in this manner it should be possible to obtain explicitly all derivatives $D^{k}\underset{\sim}{\phi}(p')$ in terms of the derivatives $D^{j}\underset{\sim}{\phi}(p')$, $j = 1,\ldots,k-1$, and partial derivatives of the functions $e_{i}(\underset{\sim}{\phi})$, $i = 1,\ldots,m + 1$, of degree up to $k + 1$.

In order that the minimizing trajectory $\underset{\sim}{\phi}(p')$ be expanded in a Taylor series about $p' = 0$, we have to show that limiting derivatives exist at $p' = 0$. For very large values of p, we can approximate (22) by

$$\begin{aligned}\underset{\sim}{\nabla}(\underset{\sim}{\nabla}U(\underset{\sim}{\phi}(p'),p))^{T}\overset{\sim}{=}q\left[\sum_{i\in K}\left(\frac{e_{ip'}}{M_{p'}}\right)^{q}\right]^{\frac{1}{q}-1}\sum_{i\in K}\Bigg\{&\left(\frac{e_{ip'}}{M_{p'}}\right)^{q-2}\left(\frac{\underset{\sim}{\nabla}e_{ip'}}{M_{p'}}\right)\\ &\cdot(\underset{\sim}{\nabla}e_{ip'})^{T}\Bigg\}=q\,\overline{H}\end{aligned}\tag{25}$$

where

$$\overline{H} \triangleq \left[\sum_{i\varepsilon K}\left(\frac{e_{ip'}}{M_{p'}}\right)^q\right]^{\frac{1}{q}-1} \sum_{i\varepsilon K}\left\{\left(\frac{e_{ip'}}{M_{p'}}\right)^{q-2}\left(\frac{\nabla \underset{\sim}{e}_{ip'}}{M_{p'}}\right)(\nabla \underset{\sim}{e}_{ip'})^T\right\} \tag{26}$$

$\overline{H}$ is an n x n matrix and under our assumptions, it is nonsingular and has an inverse. Therefore, (24) becomes

$$\begin{aligned} D\underset{\sim}{\phi}(p') &= (\operatorname{sgn} M_{p'})\,\{q\overline{H}\}^{-1}\, q^2 \left[\sum_{i\varepsilon K}\left(\frac{e_{ip'}}{M_{p'}}\right)^q\right]^{\frac{1}{q}-1} \\ &\quad \cdot \sum_{i\varepsilon K}\left\{\left(\frac{e_{ip'}}{M_{p'}}\right)^{q-1} \ln\left(\frac{e_{ip'}}{M_{p'}}\right) \nabla \underset{\sim}{e}_{ip'}\right\} \\ &= (\operatorname{sgn} M_{p'})\,\overline{H}^{-1}\left[\sum_{i\varepsilon K}\left(\frac{e_{ip'}}{M_{p'}}\right)^q\right]^{\frac{1}{q}-1} \\ &\quad \cdot \sum_{i\varepsilon K}\left\{\left(\frac{e_{ip'}}{M_{p'}}\right)^{q-1} \ln\left(\frac{e_{ip'}}{M_{p'}}\right)^q \nabla \underset{\sim}{e}_{ip'}\right\} \\ &= (\operatorname{sgn} M_{p'})\,\overline{H}^{-1}\left[\sum_{i\varepsilon K}\left(\frac{e_{ip'}}{M_{p'}}\right)^q\right]^{\frac{1}{q}} \end{aligned}$$

$$\cdot \sum_{i\varepsilon K}\left\{\frac{\left(\frac{e_{ip'}}{M_{p'}}\right)^q}{\sum_{i\varepsilon K}\left(\frac{e_{ip'}}{M_{p'}}\right)^q}\left[\ln \frac{\left(\frac{e_{ip'}}{M_{p'}}\right)^q}{\sum_{i\varepsilon K}\left(\frac{e_{ip'}}{M_{p'}}\right)^q}\right]\frac{M_{p'}}{e_{ip'}}\,\nabla \underset{\sim}{e}_{ip'}\right\}. \tag{27}$$

Let

$$s_q(p') \triangleq \left[\sum_{i\varepsilon K}\left(\frac{e_{ip'}}{M_{p'}}\right)^q\right]^{\frac{1}{q}} \tag{28}$$

$$\mu_i(p') \triangleq \frac{\left(\frac{e_{ip'}}{M_{p'}}\right)^q}{\sum_{i\varepsilon K}\left(\frac{e_{ip'}}{M_{p'}}\right)^q}. \tag{29}$$

Then

$$D\underset{\sim}{\phi}(p') = (\operatorname{sgn} M_{p'})\overline{H}^{-1}\, s_q(p') \sum_{i\varepsilon K}\{\mu_i(p')[\ln \mu_i(p')]\, \frac{M_{p'}}{e_{ip'}}\, \nabla \underset{\sim}{e}_{ip'}\}\;. \tag{30}$$

Imposing optimality conditions [11] and noting that

$$\lim_{p'\to 0} s_q(p') = 1 \tag{31}$$

$$\lim_{p'\to 0} \mu_i(p') = u_i \begin{cases} = 0 & i \notin \check{K} \triangleq \{i|e_i(\underset{\sim}{\phi}(0)) = M(\underset{\sim}{\phi}(0)), i \varepsilon I\} \\ \geq 0 & i \varepsilon \check{K} \end{cases} \tag{32}$$

$$\sum_{i\in\check{K}} u_i = 1. \tag{33}$$

The limiting value of $D\utilde{\phi}(p')$ at $p' = 0$ is therefore given by

$$D\utilde{\phi}(0) = \lim_{p'\to 0} D\utilde{\phi}(p') = (\operatorname{sgn} M(\utilde{\phi}(0)))\bar{H}^{-1} \sum_{i\in\check{K}} (u_i \ln u_i)\utilde{\nabla} e_i(\utilde{\phi}(0)). \tag{34}$$

The existence of the higher order derivatives of $\utilde{\phi}(p')$ at $p' = 0$ may be derived in a similar manner.

Consider, for example, the minimization w.r.t. $\utilde{\phi}$ of the maximum of $e_1=\phi_1^4+\phi_2^2$, $e_2 = (2-\phi_1^2)^2 + (2-\phi_2^2)^2$, $e_3 = 2\exp(-\phi_1+\phi_2)$. The solution is $\phi_1 = \phi_2 = 1$. Fig. 1 shows the Euclidean norms of $\utilde{\phi}(1/p)$ and $D\utilde{\phi}(1/p)$ as a function of $1/p$. The points correspond to minima for p in the sequence 2^i, $i = 1,\dots,14$. The limiting value of $||\utilde{\phi}(1/p)||_2$ is $\sqrt{2}$ and $||D\utilde{\phi}(1/p)||_2$ is well defined.

NUMERICAL RESULTS

Minimax Example

For the minimax example of the previous section, using the least pth objective (1) and extrapolation formula (15) for p = 4,16,64,256,1024, 52 function evaluations yielded $\phi_1 = 1.0000001$, $\phi_2 = 0.9999999$. A less accurate solution of $\phi_1 = 1.0000023$, $\phi_2 = 0.9999945$ was obtained in 62 function evaluations using $\hat{p} = 10^5$.

Beale Problem [12]

Consider the problem of minimizing

$$f(\utilde{\phi}) = 9 - 8\phi_1 - 6\phi_2 - 4\phi_3 + 2\phi_1^2 + 2\phi_2^2 + \phi_3^2 + 2\phi_1\phi_2 + 2\phi_1\phi_3$$

subject to

$$\phi_i \geq 0\ ,\ i = 1,2,3 \qquad\qquad 3 - \phi_1 - \phi_2 - 2\phi_3 \geq 0.$$

The function has a minimum $f(\check{\utilde{\phi}}) = 1/9$ at $\check{\utilde{\phi}} = [4/3\ 7/9\ 4/9]^T$. The Bandler-Charalambous technique transformed the nonlinear programming problem into an unconstrained minimax problem. The minimax problem was then solved using a sequence of least pth approximations together with extrapolation; using least pth approximation with a value of p of 10^5; using the Charalambous-Bandler algorithm [13] with a value of p of 10. Table 1 shows the results.

Parameters	p = 4,16,64,256; α = 1	$p = 10^5$	p = 10
	Order of extrapolation = 3	α = 1	α = 1
ϕ_1	1.3333333	1.3333338	1.3333334
ϕ_2	0.7777778	0.7777775	0.7777777
ϕ_3	0.4444444	0.4444437	0.4444444
$f(\utilde{\phi})$	0.1111111	0.1111114	0.1111111
Function evaluations	53	78	99

Table 1 Results for the Beale problem for starting point $\utilde{\phi}^0=[1\ \ 2\ \ 1]^T$.

LC Lowpass Filter

A symmetrical ladder network with unit resistive terminations consisting of 4 lumped lossless inductors and 3 capacitors is to be optimized to meet the following specifications. The passband (ω = 0 to 1) insertion loss Γ is not to exceed 0.01 dB while the stopband insertion loss at ω = 2.5 is to be a maximum. Table 2 compares the results obtained from using the Bandler-Charalambous minimax transformation, least pth approximation and extrapolation and from using Fiacco-McCormick SUMT method [2] with extrapolation. We find that the first approach gives better results. Discrepancies between the numerical solutions and the analytical solution are due to the finite, uniformly spaced passband points used.

Parameters	Least pth Approach	Fiacco-McCormick Method	
	$p = 4,16,64,256;\ \alpha=10^4$ Order of extrapolation=3	$r=10^{-2},2x10^{-3},\ldots,3.2x10^{-6}$ Order of extrapolation=3	Analytical Solution
ϕ_1	0.8000	0.8006	0.7969
ϕ_2	1.3929	1.3931	1.3924
ϕ_3	1.7502	1.7508	1.7481
ϕ_4	1.6332	1.6333	1.6331
ϕ_5	1.7502	1.7508	1.7481
ϕ_6	1.3929	1.3931	1.3924
ϕ_7	0.8000	0.8006	0.7969
$\Gamma(\underset{\sim}{\phi},\omega_i)_{max}$ dB	0.01003	0.01006	0.01000
$\Gamma(\underset{\sim}{\phi},\omega_s)$dB	62.96	62.98	62.87
Function evaluations	169	174	-

Table 2 Results for the LC lowpass filter design problem.

CONCLUSIONS

Theoretical considerations and some computational advantages of applying an extrapolation technique in solving minimax and related problems using a sequence of least pth approximations have been presented. Numerical results indicate that the extrapolation procedure is at least as efficient as or faster than most of the existing minimax algorithms. Charalambous has recently presented important new results in least pth optimization and nonlinear programming [14], using a different approach to acceleration.

ACKNOWLEDGEMENTS

Useful discussions are acknowledged with J.H.K. Chen, now with Bell-Northern Research, Ottawa, Canada, and with Dr. W. Kinsner and P.C. Liu of McMaster University.

REFERENCES

[1] A.V. Fiacco and G.P. McCormick, "Extensions of SUMT for nonlinear programming: Equality constraints and extrapolation", Management Sci., vol. 12, 1966, pp. 816-829.

[2] A.V. Fiacco and G.P. McCormick, Nonlinear Programming: Sequential Unconstrained Minimization Techniques. New York: Wiley, 1968, pp. 816-829.

[3] F.A. Lootsma, "Extrapolation in logarithmic programming", Philips Res. Repts. 23, 1968, pp. 108-116.

[4] G.C. Temes and D.Y.F. Zai, "Least pth approximation", IEEE Trans. Circuit Theory, vol. CT-16, May 1969, pp. 235-237.

[5] J.W. Bandler, "Optimization methods for computer-aided design", IEEE Trans. Microwave Theory Tech., vol. MTT-17, Aug. 1969, pp. 533-552.

[6] R. Seviora, M. Sablatash, and J.W. Bandler,"Least pth and minimax objectives for automated network design", Electron. Lett.,vol. 6, 1970, pp. 14-15.

[7] J.W. Bandler and C. Charalambous, "Nonlinear programming using minimax techniques", J. Optimization Theory and Applications, vol. 13, June 1974, pp. 607-619.

[8] J.W. Bandler and C. Charalambous, "Practical least pth optimization of networks", IEEE Trans. Microwave Theory Tech., vol. MTT-20, Dec. 1972, pp. 834-840.

[9] R. Fletcher, "A new approach to variable metric algorithms", Computer J., vol. 13, Aug. 1970, pp. 317-322.

[10] D.C. Joyce, "Survey of extrapolation processes in numerical analysis", SIAM Review, vol. 13, 1971, pp. 435-490.

[11] J.W. Bandler and C. Charalambous, "On conditions for optimality in least pth approximation with $p \to \infty$", J. Optimization Theory Appl., vol. 11, May 1973, pp. 556-566.

[12] J. Kowalik and M.R. Osborne, Methods for Unconstrained Optimization Problems. New York: Elsevier, 1968.

[13] C. Charalambous and J.W. Bandler, "Nonlinear minimax optimization as a sequence of least pth optimization with finite values of p", Faculty of Engineering, McMaster University, Hamilton, Canada, June 1973, Report SOC-3.

[14] C. Charalambous, "Nonlinear least pth optimization and nonlinear programming", Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Canada, Jan. 1974, Research Report CORR 74-3.

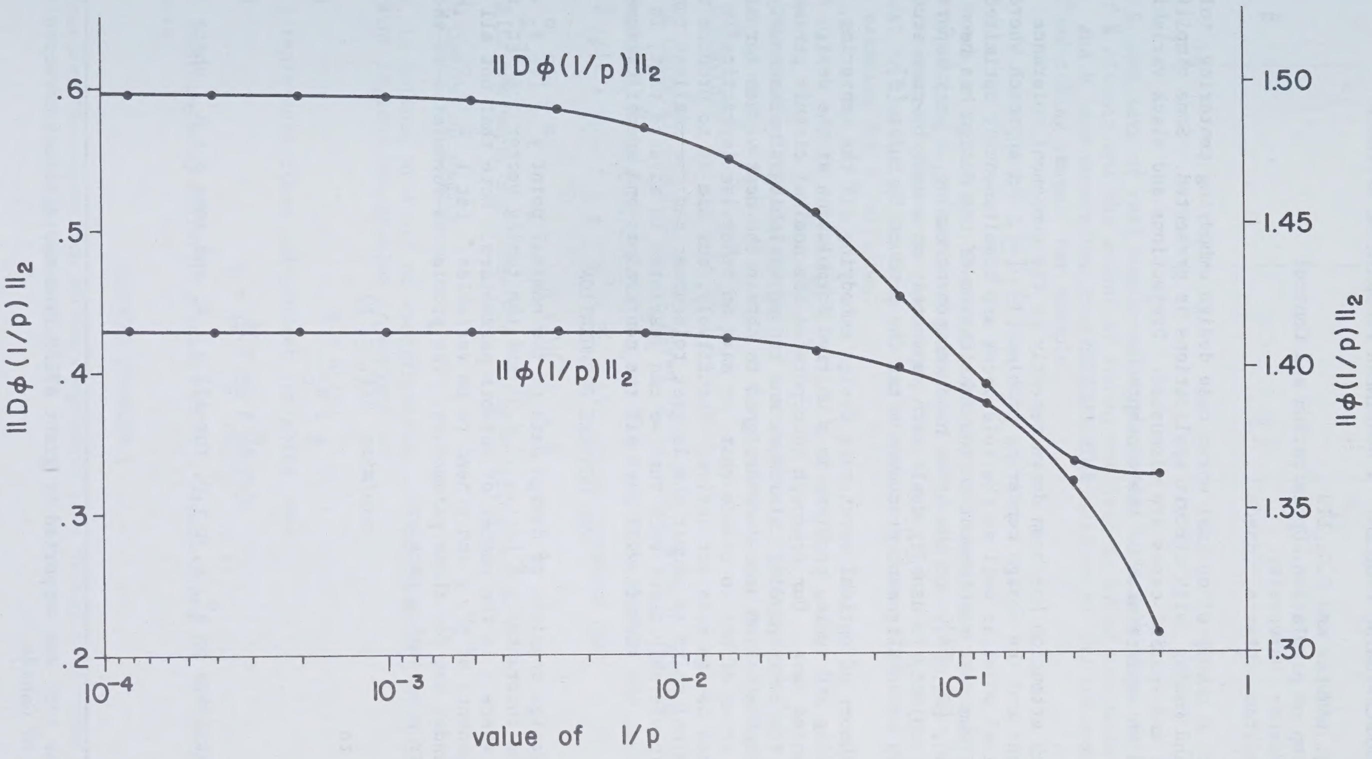


Fig. 1 $||\underset{\sim}{\phi}(\frac{1}{p})||_2$ and $||D\underset{\sim}{\phi}(\frac{1}{p})||_2$ as a function of $\frac{1}{p}$.

THE TOLERANCE-TUNING PROBLEM: A NONLINEAR PROGRAMMING APPROACH

J.W. BANDLER and P.C. LIU
Group on Simulation, Optimization and Control
McMaster University
Hamilton, Ontario, Canada

Abstract A theory of optimal worst case design embodying centering, tolerancing and tuning with circuit applications is presented. Some simplified problems and special cases are discussed. Projections and slack variables are used to explain some of the concepts.

INTRODUCTION

Much attention has been drawn recently to the component tolerance assignment and the design centering problem [1] - [4]. An approach whereby the nominal point as well as the tolerances are simultaneously optimized to meet minimum cost requirement in the fabrication of the design has been proposed [1], [2], [4]. On the other hand, component tuning, a very important related subject, is usually dealt with separately on a case-by-case study, or merely as an alignment procedure after the product is built [5].

A theory of optimal worst-case design embodying all the centering, tolerancing and tuning problems in a unified formulation at the design stage is presented here. Our approach incorporates the nominal circuit parameter values, the corresponding tolerances and tuning variables simultaneously into an optimization procedure designed to obtain the best values for all of them in an effort to reduce cost, or make an otherwise impractically toleranced design more attractive. Intuitively, our aim is to produce the best nominal point to permit the largest tolerances and the smallest tuning ranges (preferably zero) such that we can guarantee in advance that, in the worst case, the circuit will meet all the constraints and specifications.

THE GENERAL FORMULATION

A design consists of design data of the nominal point $\underset{\sim}{\phi}^o \triangleq [\phi_1^o\ \phi_2^o \ldots \phi_k^o]^T$, the tolerances $\underset{\sim}{\varepsilon} \triangleq [\varepsilon_1\ \varepsilon_2 \ldots \varepsilon_k]^T$ and the tuning vector $\underset{\sim}{t} \triangleq [t_1\ t_2 \ldots t_k]^T$, where k is the number of network parameters. Note that not all the components of $\underset{\sim}{\phi}^o$, $\underset{\sim}{\varepsilon}$ and $\underset{\sim}{t}$ need to be variables. Let $I_\phi \triangleq \{1,2,\ldots,k\}$ be the index set for these parameters. The problem is formulated as the nonlinear programming problem:

$$\text{minimize} \quad C(\underset{\sim}{\phi}, \underset{\sim}{\varepsilon}, \underset{\sim}{t})$$

subject to

$$\underset{\sim}{\phi} \in R_c \tag{1}$$

where

$$\underset{\sim}{\phi} = \underset{\sim}{\phi}^o + \underset{\approx}{E}\underset{\sim}{\mu} + \underset{\approx}{T}\underset{\sim}{\rho} \tag{2}$$

and constraints on $\underset{\sim}{\phi}^o$, $\underset{\sim}{\varepsilon}$, $\underset{\sim}{t} \geq \underset{\sim}{0}$, for all $\underset{\sim}{\mu} \in R_\mu$ and some $\underset{\sim}{\rho} \in R_\rho$, where

---

This work was supported by grant A7239 from the National Research Council of Canada.

$$\boldsymbol{E} \triangleq \begin{bmatrix} \varepsilon_1 & & & & \\ & \varepsilon_2 & & & \\ & & \cdot & & \\ & & & \cdot & \\ & & & & \cdot \\ & & & & & \varepsilon_k \end{bmatrix}, \qquad \boldsymbol{T} \triangleq \begin{bmatrix} t_1 & & & & \\ & t_2 & & & \\ & & \cdot & & \\ & & & \cdot & \\ & & & & \cdot \\ & & & & & t_k \end{bmatrix}.$$

R_μ and R_ρ are sets of real number multipliers that will determine the outcome of a circuit and the amount of tuning required to bring the point into R_c. R_μ and R_ρ are determined from realistic situations of the tolerance spread and tuning range. For example,

$$R_\mu = \{\boldsymbol{\mu} \mid -1 \leq \mu_i \leq -a_i,\ a_i \leq \mu_i \leq 1,\ i \in I_\phi\} \tag{3}$$

where $0 \leq a_i \leq 1$, $i \in I_\phi$. The most commonly used continuous range is obtained by setting a_i to zero. A commercial stock will probably have the better toleranced components taken out, thus $0 < a_i < 1$. If $a_i = 1$, R_μ is identical to the set of vertices of the tolerance region. Some of the common examples for R_ρ will be,

$$R_\rho = \{\boldsymbol{\rho} \mid -1 \leq \rho_i \leq 1,\ i \in I_\phi\} \tag{4}$$

or, in the case of one-way tuning or irreversible trimming, $R_\rho = \{\boldsymbol{\rho} \mid 0 \leq \rho_i \leq 1,\ i \in I_\phi\}$, or $R_\rho = \{\boldsymbol{\rho} \mid -1 \leq \rho_i \leq 0,\ i \in I_\phi\}$. Unless otherwise indicated, we will consider $-1 \leq \mu_i \leq 1$, $-1 \leq \rho_i \leq 1$, $i \in I_\phi$.

The constraint region R_c is given by $R_c \triangleq \{\boldsymbol{\phi} \mid \boldsymbol{g}(\boldsymbol{\phi}) \geq \boldsymbol{0}\}$ and the tolerance region $R_\varepsilon \triangleq \{\boldsymbol{\phi} \mid \boldsymbol{\phi}^o - \boldsymbol{\varepsilon} \leq \boldsymbol{\phi} \leq \boldsymbol{\phi}^o + \boldsymbol{\varepsilon}\}$. Thus, a tuning region $R_t(\boldsymbol{\mu}) \triangleq \{\boldsymbol{\phi} \mid \boldsymbol{\phi}^o + \boldsymbol{E\mu} - \boldsymbol{t} \leq \boldsymbol{\phi} \leq \boldsymbol{\phi}^o + \boldsymbol{E\mu} + \boldsymbol{t}\}$. It is required that

$$R_t(\boldsymbol{\mu}) \cap R_c \neq \emptyset, \tag{5}$$

where $\emptyset$ is the empty set. See Fig. 1 for an illustration.

Tunable Constraint Region

R_c is defined by a set of specifications and constraints. For tunable constraint regions the problem is subject to

$$\boldsymbol{\phi} \in R_c(\boldsymbol{\psi}) \tag{6}$$

where $\boldsymbol{\psi}$ represents other independent variables, and

$$\boldsymbol{\phi} = \boldsymbol{\phi}^o + \boldsymbol{E\mu} + \boldsymbol{T\rho}(\boldsymbol{\psi}) \tag{7}$$

Fig. 2 depicts three different regions of R_c. Overlapping is, of course, allowable.

REDUCED PROBLEM 1

Consider the separation of the components into effectively tuned and effectively toleranced parameters. Let

$$I_\varepsilon \triangleq \{i | \varepsilon_i > t_i,\ i\ \varepsilon\ I_\phi\} \tag{8}$$

$$I_t \triangleq \{i | t_i \geq \varepsilon_i,\ i\ \varepsilon\ I_\phi\} \tag{9}$$

$$\varepsilon_i' \triangleq \varepsilon_i - t_i\ ,\ i\ \varepsilon\ I_\varepsilon \tag{10}$$

and

$$t_i' \triangleq t_i - \varepsilon_i,\ i\ \varepsilon\ I_t . \tag{11}$$

It is obvious that I_t and I_ε are disjoint and $I_t \bigcup I_\varepsilon = I_\phi$. Now, we may consider the problem subject to

$$\underset{\sim}{\phi}\ \varepsilon\ R_c$$

where

$$\phi_i = \phi_i^o + \begin{cases} \varepsilon_i'\ \mu_i & \text{for } i\ \varepsilon\ I_\varepsilon \\ t_i'\ \rho_i' & \text{for } i\ \varepsilon\ I_t \end{cases} \tag{12}$$

for all $-1 \leq \mu_i \leq 1$, $i\ \varepsilon\ I_\varepsilon$ and for some $-1 \leq \rho_i' \leq 1$, $i\ \varepsilon\ I_t$.

Theorem 1

A feasible solution to the reduced problem 1 is a feasible solution to the original problem.

Proof

Given $\underset{\sim}{\phi}^o$, $\underset{\sim}{\varepsilon}$, $\underset{\sim}{t}$ we will show that

(1) $\varepsilon_i\ \mu_i + t_i\ \rho_i = \varepsilon_i'\ \mu_i \qquad i\ \varepsilon\ I_\varepsilon$

(2) $\varepsilon_i\ \mu_i + t_i\ \rho_i = t_i'\ \rho_i' \qquad i\ \varepsilon\ I_t$

under the restrictions on μ_i, ρ_i and ρ_i'.

(1) Since ρ_i can be freely chosen from $-1 \leq \rho_i \leq 1$, we can let $\rho_i = -\mu_i$ giving

$$(\varepsilon_i - t_i)\mu_i = \varepsilon_i'\ \mu_i .$$

(2) For any $-1 \leq \rho_i' \leq 1$ and all $-1 \leq \mu_i \leq 1$, we can choose

$$-1 \leq \rho_i = \frac{(t_i - \varepsilon_i)\rho_i' - \varepsilon_i \mu_i}{t_i} \leq 1.$$

Thus, any point with components represented by (12) of the reduced problem can be represented by (2) of the original problem.

We state here without proof the following theorem.

Theorem 2

A feasible solution to the original problem implies a feasible solution to the reduced problem if R_c is one-dimensionally convex.

A GEOMETRIC INTERPRETATION

Let

$$R_{\epsilon t} \triangleq \{\underset{\sim}{\phi} \mid \phi_i^o - \epsilon_i' \leq \phi_i \leq \phi_i^o + \epsilon_i',\ i \ \epsilon \ I_\epsilon\} \tag{13}$$

and

$$R_{t\epsilon} \triangleq \{\underset{\sim}{\phi} \mid \phi_i^o - t_i' \leq \phi_i \leq \phi_i^o + t_i',\ i \ \epsilon \ I_t\} . \tag{14}$$

Consider the following regions

$$R_{\epsilon tp} \triangleq \{\underset{\sim}{\phi}_p \mid \underset{\sim}{\phi}_p = \underset{\sim}{P}\underset{\sim}{\phi},\ \underset{\sim}{\phi} \ \epsilon \ R_{\epsilon t}\} , \tag{15}$$

$$R_{ct\epsilon} \triangleq R_c \cap R_{t\epsilon}$$

and

$$R_{ct\epsilon p} \triangleq \{\underset{\sim}{\phi}_p \mid \underset{\sim}{\phi}_p = \underset{\sim}{P}\underset{\sim}{\phi},\ \underset{\sim}{\phi} \ \epsilon \ R_{ct\epsilon}\} \tag{16}$$

where

$$\underset{\sim}{P} \triangleq \begin{bmatrix} p_1 & & & & \\ & p_2 & & & \\ & & \cdot & & \\ & & & \cdot & \\ & & & & \cdot \ \ p_k \end{bmatrix}$$

and

$$p_i = \begin{cases} 0 \text{ for } & i \ \epsilon \ I_t \\ 1 \text{ for } & i \ \epsilon \ I_\epsilon \end{cases}$$

The constraints of the problem are now interpreted by the requirement that

$$R_{\epsilon tp} \subseteq R_{ct\epsilon p} . \tag{17}$$

Fig. 3 depicts the definition of the different regions. Any point whose components are given by (12) is in the intersection of $R_{\epsilon t}$ and $R_{t\epsilon}$, may be tuned into $R_{ct\epsilon}$ by changing the value of ρ_i, $i \ \epsilon \ I_t$ if the projection of $R_{ct\epsilon}$ onto the subspace spanned by the components subject to tolerances includes the projection of that point.

The reduced problem may be stated as: solve a pure tolerance problem (i.e., no tuning) in the subspace spanned by the toleranced variables with $R_{\epsilon tp}$ as the tolerance region and $R_{ct\epsilon p}$ as the constraint region.

Special Cases

(1) $I_\epsilon = \emptyset$, the pure tuning problem.
In this case, $R_{\epsilon t}$ is the entire space and $\underset{\sim}{P}$ is a zero matrix. The problem has a solution if

$$R_{ct\epsilon} \neq \emptyset . \tag{18}$$

(2) $I_t = \emptyset$, the pure tolerance problem.
In this case, $R_{t\varepsilon}$ is the entire space and $\underset{\sim}{P}$ is a unit matrix. This problem has been treated thoroughly. It is required that,

$$R_{\varepsilon t} \subseteq R_c \quad . \tag{19}$$

Fig. 4 illustrates a case where $R_{\varepsilon tp} \not\subseteq R_{ct\varepsilon p}$. An outcome at $\underset{\sim}{\phi}^o$ cannot be tuned to R_c within the effective tuning range. However, there exists a solution to the original formulation by tuning both components. R_c is not one-dimensionally convex in this case.

Extension for Tunable Constraint Region

Three types of components can be identified when the constraint region is considered to be tunable. They are:

(a) Fixed components with tolerances
(b) Components tuned by the manufacturer
(c) Components tunable by the customer

In this case,

$$\underset{\sim}{\phi} \in R_c(\underset{\sim}{\psi})$$

where

$$\phi_i = \phi_i^o + \begin{cases} \varepsilon_i' \mu_i & \text{for } i \in I_\varepsilon \\ t_i' \rho_i' & \text{for } i \in I_{tm} \\ t_i' \rho_i'(\underset{\sim}{\psi}) & \text{for } i \in I_{tc} \end{cases} \tag{20}$$

where I_{tm} identifies components (b) and I_{tc} identifies components (c).

Setting the value $\underset{\sim}{\psi}$ to a particular value will control the setting of the value of ρ_i', $i \in I_{tc}$ such that $\underset{\sim}{\phi}$ will be in that particular constraint region $R_c(\underset{\sim}{\psi})$.

REDUCED PROBLEM 2

We consider $-1 \leq \mu_i \leq 1$ replaced by $\mu_i \in \{-1,1\}$, $i \in I_\varepsilon$.

Theorem 3 A feasible solution to reduced problem 2 implies a feasible solution to reduced problem 1 if $R_{ct\varepsilon p}$ is one-dimensionally convex.

This is a pure tolerance problem in the subspace. The proof is given in [1].

Implementation for Frequency Domain Problem

Let $\underset{\sim}{\phi}_p^r$ be a vertex of $R_{\varepsilon tp}$. The constraints can be expressed explicitly as

$$\underset{\sim}{g}(\underset{\sim}{\phi}) = \underset{\sim}{g}(\underset{\sim}{\phi}_p^r + \sum_{i \in I_t} (\phi_i^o + t_i' \rho_i^r)\underset{\sim}{e}_i) \geq \underset{\sim}{0} \tag{21}$$

where $\underset{\sim}{e}_i$ is the ith unit vector. The slack variables ρ_i^r are also to be constrained by $-1 \leq \rho_i^r \leq 1$. Strategies to eliminate inactive vertices [4] will not be discussed here.

The optimization parameters $\underset{\sim}{x}$ may now be identified as a n-dimensional vector consisting of the variable nominal values, tolerances, tuning variables and all the appropriate slack variables $\underset{\sim}{\rho}$. A total of m constraint functions may be formed. in general,

$$n = k_o + k_\varepsilon + k_t(i + n_v) \qquad (22)$$

and

$$m = \left(\sum_{i=1}^{n_\omega} n_v(i) \right) + 2k_t n_v + \ldots \qquad (23)$$

where k_o, k_ε and k_t are the numbers of nominal parameter values, toleranced and tuned parameters, respectively; $n_v \leq 2^{k_\varepsilon}$ is the number of distinct vertices chosen; n_ω is the number of frequency points considered; $n_v(i)$ is the number of vertices chosen at the ith frequency point and $2k_t n_v$ is the number of slack variable bounds.

Any optimization algorithm may be used to minimize $C(\underset{\sim}{x})$ subject to $g_i(\underset{\sim}{x}) \geq 0$, $i = 1,2,\ldots,m$.

EXAMPLE

Consider the constraints

$$\phi_2 - \phi_1 - 2 \geq 0 \qquad (24)$$

$$-\phi_2^2 + 16\phi_1 \geq 0 \,. \qquad (25)$$

Tolerance Example

Minimize $\dfrac{1}{\varepsilon_1} + \dfrac{1}{\varepsilon_2}$

subject to

$$g_1 = \varepsilon_1 \geq 0 \;,\; g_2 = \varepsilon_2 \geq 0,\; g_3 = \phi_1^o \geq 0,\; g_4 = \phi_2^o \geq 0$$

$$g_5(i) = (\phi_2^o + \varepsilon_2\mu_2(i)) - (\phi_1^o + \varepsilon_1\mu_1(i)) - 2 \geq 0 \quad i = 1,2,\ldots \qquad (26)$$

$$g_6(i) = -(\phi_2^o + \varepsilon_2\mu_2(i))^2 + 16(\phi_1^o + \varepsilon_1\mu_1(i)) \geq 0 \quad i = 1,2,\ldots \qquad (27)$$

where $-1 \leq \mu_1(i) \leq 1$ and $-1 \leq \mu_2(i) \leq 1$.

Optimality requires

$$\begin{bmatrix} -\dfrac{1}{\varepsilon_1^2} \\ -\dfrac{1}{\varepsilon_2^2} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} + \sum_i u_5(i) \begin{bmatrix} -\mu_1(i) \\ \mu_2(i) \\ -1 \\ 1 \end{bmatrix} + \sum_i u_6(i) \begin{bmatrix} 16\mu_1(i) \\ -2\mu_2(i)\,(\phi_2^o + \varepsilon_2\mu_2(i)) \\ 16 \\ -2(\phi_2^o + \varepsilon_2\mu_2(i)) \end{bmatrix} \qquad (28)$$

$$u_1 g_1 = \ldots = u_4 g_4 = u_5(i)\, g_5(i) = u_6(i)\, g_6(i) = 0 \quad i = 1,2,\ldots \qquad (29)$$

$$u_1, \ldots, u_4, u_5(i), u_6(i) \geq 0 \qquad i = 1,2,\ldots \qquad (30)$$

Assume that ε_1, ε_2, ϕ_1^o and ϕ_2^o are not zero, therefore, set u_1, u_2, u_3 and u_4 to zero. Minimize $g_5(i)$ of (26) and $g_6(i)$ of (27) w.r.t. $\underset{\sim}{\mu}(i)$. This leads, respectively, to

$$(\phi_2^o - \varepsilon_2) - (\phi_1^o + \varepsilon_1) - 2 \geq 0$$

using $\underset{\sim}{\mu}(i) = [1 \;\; -1]^T$ and

$$-(\phi_2^o + \varepsilon_2)^2 + 16(\phi_1^o - \varepsilon_1) \geq 0$$

using $\underset{\sim}{\mu}(i) = [-1 \quad 1]^T$. The optimality conditions (28) - (30) are correspondingly reduced yielding the solution

$$\varepsilon_1 = 0.5 \ , \ \varepsilon_2 = 0.5 \ , \ \phi_1^o = 3.5 \ , \ \phi_2^o = 7.5 \ ,$$

which have been verified by numerical optimization.

Tuning Example

Suppose ϕ_1 has a 10% tuning range and ϕ_2 is toleranced. Consider the problem of minimizing

$$\frac{1}{\varepsilon_2'}$$

subject to

$$g_1 = t_1' \geq 0, \ g_2 = \varepsilon_2' \geq 0, \ g_3 = \phi_1^o \geq 0, \ g_4 = \phi_2^o \geq 0$$

$$g_5 = .1 - \frac{t_1'}{\phi_1^o} \geq 0 \tag{31}$$

$$g_6(i) = (\phi_2^o + \varepsilon_2'\mu_2(i)) - (\phi_1^o + t_1'\rho_1'(i)) - 2 \geq 0 \quad i = 1,2,\ldots \tag{32}$$

$$g_7(i) = -(\phi_2^o + \varepsilon_2'\mu_2(i))^2 + 16(\phi_1^o + t_1'\rho_1'(i)) \geq 0 \quad i = 1,2,\ldots \tag{33}$$

$$g_8(i) = 1 - \rho_1'(i) \geq 0 \quad i = 1,2,\ldots \tag{34}$$

$$g_9(i) = 1 + \rho_1'(i) \geq 0 \quad i = 1,2,\ldots \tag{35}$$

The variables are t_1', ε_2', ϕ_1^o, ϕ_2^o and $\rho_1'(i)$, and $-1 \leq \mu_2(i) \leq 1$.

Optimality requires

$$\begin{bmatrix} 0 \\ -\frac{1}{\varepsilon_2'^2} \\ 0 \\ 0 \\ \underset{\sim}{0} \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ \underset{\sim}{0} \end{bmatrix} + u_5 \begin{bmatrix} -\frac{1}{\phi_1^o} \\ 0 \\ \frac{t_1'}{\phi_1^{o2}} \\ 0 \\ \underset{\sim}{0} \end{bmatrix} + \sum_i u_6(i) \begin{bmatrix} -\rho_1'(i) \\ \mu_2(i) \\ -1 \\ 1 \\ -t_1'\underset{\sim}{e}_i \end{bmatrix} + \sum_i u_7(i) \begin{bmatrix} 16\rho_1'(i) \\ -2(\phi_2^o+\varepsilon_2'\mu_2(i))\mu_2(i) \\ 16 \\ -2(\phi_2^o+\varepsilon_2'\mu_2(i)) \\ 16t_1'\underset{\sim}{e}_i \end{bmatrix}$$

$$+ \sum_i u_8(i) \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -\underset{\sim}{e}_i \end{bmatrix} + \sum_i u_9(i) \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \underset{\sim}{e}_i \end{bmatrix} \tag{36}$$

$$u_1 g_1 = \ldots = u_5 g_5 = u_6(i)\ g_6(i) = \ldots = u_9(i)\ g_9(i) = 0 \quad i = 1,2,\ldots \tag{37}$$

$$u_1,\ldots,u_5,\ u_6(i),\ldots,u_9(i) \geq 0 \quad i = 1,2,\ldots \tag{38}$$

where $\underset{\sim}{e}_i$ is the ith unit vector. Minimize $g_6(i)$ of (32) and $g_7(i)$ of (33) w.r.t. $\mu_2(i)$. We use $\mu_2(i) = -1$ in (32) and $\mu_2(i) = 1$ in (33) for this purpose. The corresponding $\rho_1'(i) = -1$ and $\rho_1'(i) = 1$, respectively, are obtained by maximizing $g_6(i)$ and $g_7(i)$ w.r.t. $\rho_1'(i)$. This yields the solution $t_1' = 0.5432$, $\varepsilon_2' = 1.444$, $\phi_1^o = 5.4321$, $\phi_2^o = 8.3333$.

CONCLUSIONS

A theory of optimal worst case design embodying centering, tolerancing and tuning with circuit applications has been presented. The concept of a tunable constraint region that allows variable specifications as set by the customer has also been incorporated. This may find application, for example, in tunable filters. Components can be separated into effectively tuned and effectively toleranced parameters to simplify the solution, but possibly at the expense of optimality. The purely toleranced and purely tuned problems become special cases. Further simplification has been discussed in the light of one-dimensional convexity.

It may be added that as far as the authors are aware, this seems to be the most general formulation to date dealing with the centering, tolerancing and tuning problems at the design stage. Tuning uncertainties can also be taken care of in the formulation.

REFERENCES

[1] J.W. Bandler, "Optimization of design tolerances using nonlinear programming", J. Optimization Theory and Applications, vol. 14, July 1974, pp. 99-114.

[2] J.W. Bandler and P.C. Liu, "Automated network design with optimal tolerances", IEEE Trans. Circuit and Systems, vol. CAS-21, Mar. 1974, pp. 219-222.

[3] J.F. Pinel and K.A. Roberts, "Tolerance assignment in linear networks using nonlinear programming", IEEE Trans. Circuit Theory, vol. CT-19, Sept. 1972, pp. 475-479.

[4] J.W. Bandler, P.C. Liu and J.H.K. Chen, "Worst case network tolerance optimization", Faculty of Engineering, McMaster Univ., Hamilton, Ont., Canada, Rep. SOC-49, Aug. 1974.

[5] J.F. Pinel, "Computer-aided network tuning", IEEE Trans. Circuit Theory, vol. CT-18, Jan. 1971, pp. 192-194.

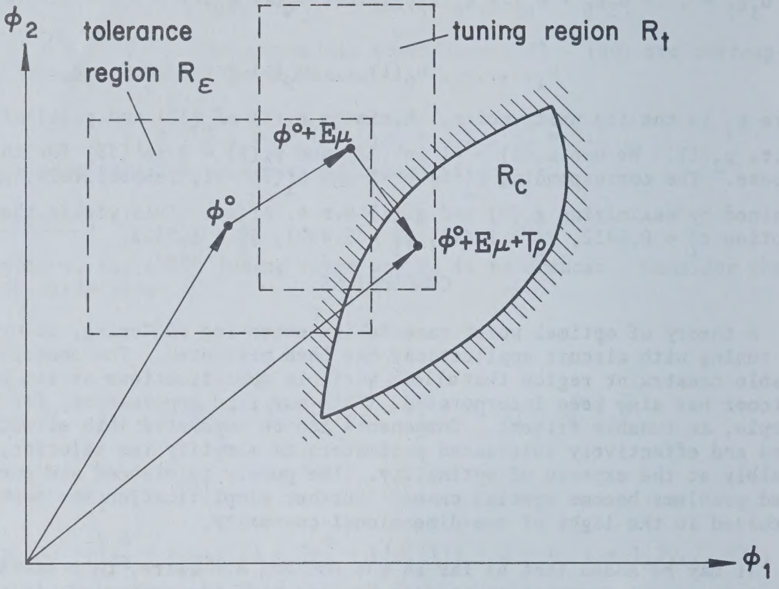

ϕ_2
tolerance
region R_ε
tuning region R_t
$\phi^\circ + E\mu$
ϕ°
R_c
$\phi^\circ + E\mu + T\rho$
ϕ_1

Fig. 1 An illustration of the regions R_ε, R_t, and R_c.

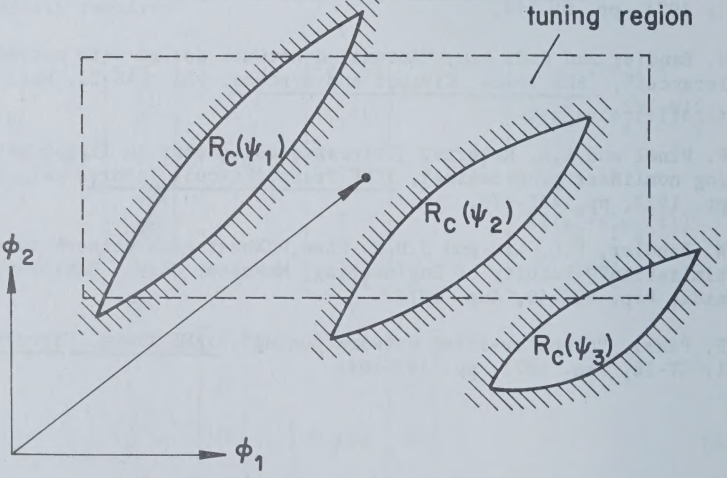

tuning region
$R_c(\psi_1)$
$R_c(\psi_2)$
$R_c(\psi_3)$
ϕ_2
ϕ_1

Fig. 2 An example of three different settings of the tunable constraint regions.

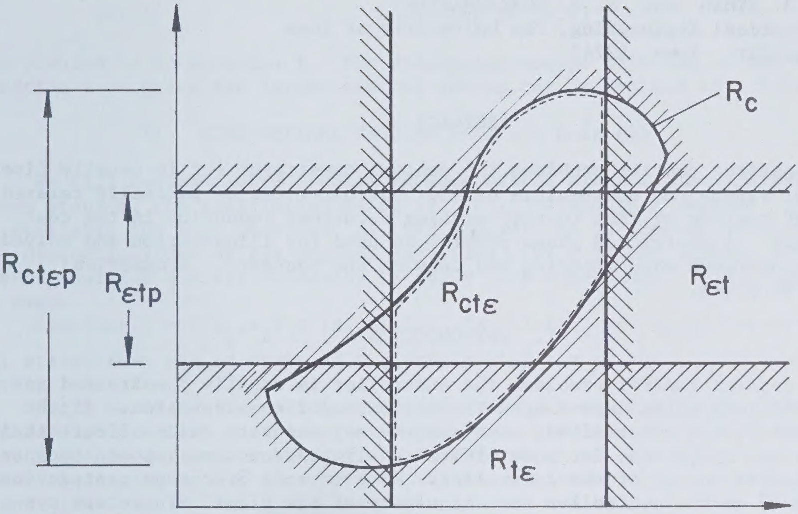

Rc
Rctεp
Rεtp
Rctε
Rεt
Rtε

Fig. 3 A geometric interpretation of the reduced problem 1.

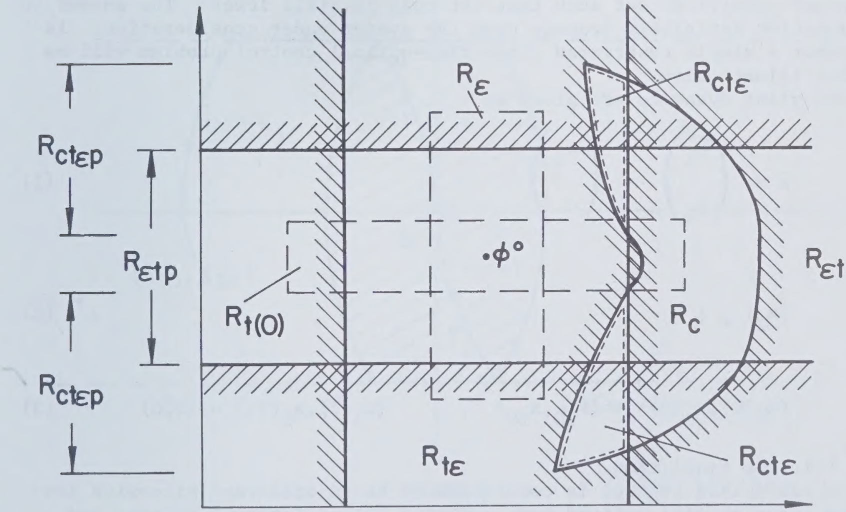

Rε
Rctε
Rctεp
Rεtp
•φ°
Rt(0)
Rc
Rεt
Rctεp
Rtε
Rctε

Fig. 4 An example of $R_{\varepsilon tp} \not\subseteq R_{ct\varepsilon p}$.

OPTIMAL CONTROL WITH TRANSITORILY ENLARGED CONTROL CONSTRAINT SETS

E. D. EYMAN and M. B. SUBRAHMANYAM
Electrical Engineering, The University of Iowa
Iowa City, Iowa 52242

ABSTRACT

In optimal control problems the control constraint set is usually fixed. For some systems the constraints on the controller can be partially relaxed for short periods of time thereby causing a further reduction in the cost functional. A restricted phase problem is used for illustration and solved by a new approach when entering and leaving the boundary. A numerical example is given.

I. INTRODUCTION

In optimal control problems the controller is usually constrained and the constraints often depend upon the system under consideration. If the controller is not constrained, some systems may saturate which affects their dynamics and operation; for some others the limitations are imposed because of the finite energy of the controller. Also in some processes restrictions are imposed on the controller for the safety of the plant. There are systems where the control constraint set can be larger for short periods of time without affecting the safety or dynamics of the plant. For example, the voltage applied to a motor can be higher than its nominal value for a short period of time without affecting the insulation properties of the windings and without overheating. In time-optimal problems with 'outage' (short period of time during which the controller is identically zero due to some malfunction [1]), it may even become necessary to make use of the larger constraint set. Now in general, the question is, can we make use of the larger constraint set such that the cost is still lower? The answer to this question definitely depends upon the system under consideration. In this paper a simple restricted phase time-optimal control problem will be used for illustration.

The plant dynamics are given by

$$\dot{x} = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ u \end{pmatrix} \tag{1}$$

with

$$|x_2| \leq 1 \tag{2}$$

and

$$(x_1(0), x_2(0)) = (x_{10}, x_{20}) \quad , \qquad (x_1(T), x_2(T)) = (0,0) \tag{3}$$

where T is the final time.

An admissible control is considered to be an arbitrary piecewise continuous function u(t) defined over a finite closed interval of time and which is continuous at both end points of the interval. Let $[\tau_1, \tau_2] \subset [0,T]$ be a closed interval of Lebesgue measure τ, and let $\mu \geq 0$ be given; $\mu << 1$, $\tau << T$. Let

$$\begin{aligned} |u(t)| &\leq 1 + \mu \qquad && t \epsilon [\tau_1, \tau_2] \\ |u(t)| &\leq 1 && \text{elsewhere on } [0,T] \quad . \end{aligned} \tag{4}$$

The problem is to minimize T. For simplicity assume only one period of duration τ on which the larger control action can be applied if $\mu > 0$.

II. TIME-OPTIMAL PROBLEM WITH $\mu = 0$ in (4)

We will describe this problem in some detail since it is a restricted phase problem, the restriction being $|x_2| \leq 1$. Although a maximum principle is available for such problems [2], its application is limited because of the discontinuities of the adjoint variables, the a priori unknown number of discontinuities and the necessity to apply jump conditions. A new approach is taken.

Consider $G = \{(x_1,x_2) : |x_2| \leq 1\}$. In G let Γ_- and Γ_+ be the sets of all states that can be drawn to the origin with $u = -1$ and $u = +1$ respectively. Let $\Gamma = \Gamma_- \cup \Gamma_+$.

Let B be the set of all points in G for which the restricted phase problem is identical to the unrestricted phase problem. B is indicated in Fig. 1 and is a closed subset of G. In this paper we consider only the points belonging to $G \cap B^c$, where B^c is the complement of B in the phase plane. In G let R_- and R_+ be the regions above and below Γ. Let ∂G denote the boundary of G. If a segment of the optimal trajectory lies in ∂G, then on this segment $u = 0$ since on the boundary $x_2 = \pm 1$ and $u = \dot{x}_2 = 0$. Let $\partial G_1 = R_+ \cap \{(x_1,x_2) : x_2 = -1\}$ and $\partial G_2 = R_- \cap \{(x_1,x_2) : x_2 = +1\}$.

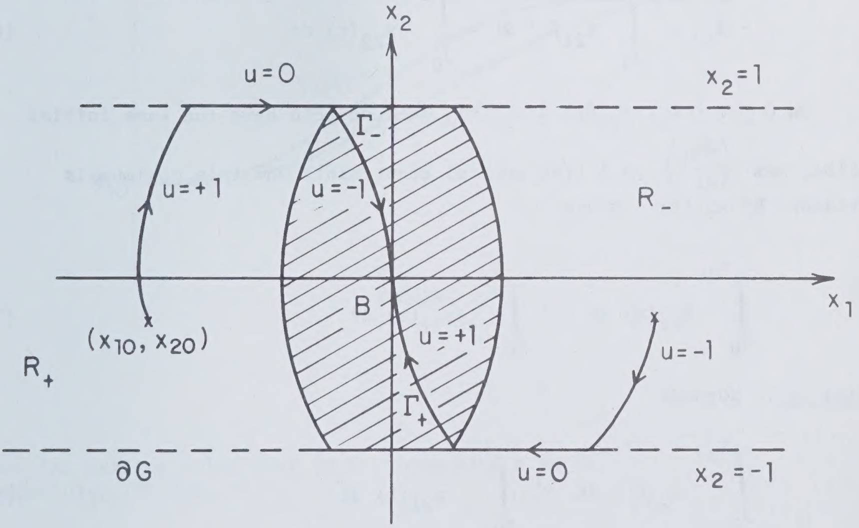


FIGURE 1

Theorem 1: The optimal control as a function of the state is given by the following control law:

$$u(x_1,x_2) = \begin{cases} 1 & (x_1,x_2) \in \Gamma_+ \cup (R_+ - \partial G) \cup \partial G_1 \\ 0 & (x_1,x_2) \in \partial G - (\partial G_1 \cup \partial G_2) \\ -1 & (x_1,x_2) \in \Gamma_- \cup (R_- - \partial G) \cup \partial G_2 \end{cases}$$

Proof: We will only consider points belonging to $G \cap B^c$. The theorem is true for points belonging to B also [2]. Let (x_{10},x_{20}) be the initial state and let $(x_{11}(t),x_{21}(t))$ be the response starting at (x_{10},x_{20}) and reaching the origin under the control given by (5). This is clearly possible.

A. Let $x_{10} < 0$. We apply $u = +1$ until the response starting at (x_{10},x_{20}) hits the boundary in the second quadrant. From there the state of the system is driven to the point of intersection of Γ_- and ∂G under $u = 0$ and thereafter $u = -1$ is applied to bring the state to the origin along Γ_-. Let u_1 denote the control strategy which accomplishes this. We have to show that u_1 is time-optimal. $x_{11}(t)$ and $x_{21}(t)$ are sketched in Fig. 2. Let t_1 be the time at which the response first enters the boundary, t_2 be the exit time and T be the time at which it reaches the origin. Clearly on $[0,t_2]$, $x_{21}(t)$ is nondecreasing.

Consider any other trajectory $(x_{12}(t),x_{22}(t))$ which has the same initial and terminal conditions as $(x_{11}(t),x_{12}(t))$ and u_2 be the corresponding control. Let $(x_{12}(t),x_{22}(t))$ reach the origin at time T' starting at (x_{10},x_{20}) at zero time. Then the following relation should hold:

$$- x_{10} = \int_0^T x_{21}(t)\, dt = \int_0^{T'} x_{22}(t)\, dt \tag{6}$$

On $0 \le t \le t_1$, $x_{22}(t) \le x_{21}(t)$, because both have the same initial value, $\max \left(\frac{dx_2}{dt}\right) = 1$ (the control constraint) and this maximum is attained by $x_{21}(t)$. Hence

$$\int_0^{t_1} x_{22}(t)\, dt \le \int_0^{t_1} x_{21}(t)\, dt \tag{7}$$

Case (a): Suppose

$$\int_0^{t_1} x_{22}(t)\, dt < \int_0^{t_1} x_{21}(t)\, dt \tag{8}$$

clearly

$$\int_{t_1}^{t_2} x_{22}(t)\, dt \le \int_{t_1}^{t_2} x_{21}(t)\, dt \tag{9}$$

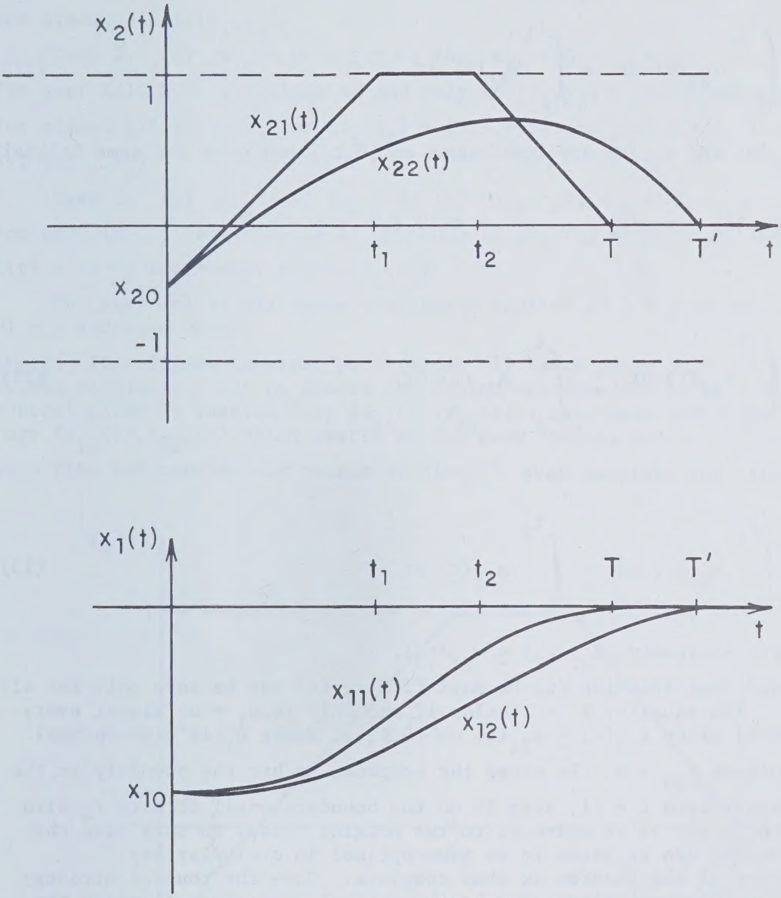


FIGURE 2

because the maximum value that $|x_2(t)|$ can attain is 1.

Combining (8) and (9),

$$\int_0^{t_2} x_{22}(t)\, dt < \int_0^{t_2} x_{21}(t)\, dt \tag{10}$$

Also, $x_{22}(t_2) \leq x_{21}(t_2)$.

In order that relation (6) be satisfied, $x_{22}(t)$ can be zero only for all $t \geq T' > T$. This implies that u_1 is time-optimal.

Case (b): Suppose

$$\int_0^{t_1} x_{22}(t)\,dt = \int_0^{t_1} x_{21}(t)\,dt \tag{11}$$

Because $x_{22}(t)$ and $x_{21}(t)$ are continuous on $[0,t_1]$ and have the same initial value

$$x_{22}(t) = x_{21}(t) \qquad t\varepsilon[0,t_1]$$

Clearly

$$\int_{t_1}^{t_2} x_{22}(t)\,dt \leq \int_{t_1}^{t_2} x_{21}(t)\,dt \tag{12}$$

Combining (11) and (12), we have

$$\int_0^{t_2} x_{22}(t)\,dt \leq \int_0^{t_2} x_{21}(t)\,dt \tag{13}$$

and moreover, obviously, $x_{22}(t_2) \leq x_{21}(t_2)$.

In order that relation (6) be satisfied $x_{22}(t)$ can be zero only for all $t \geq T' \geq T$. The equality $T' = T$ holds if and only if $u_1 = u_2$ almost everywhere on $[0,T]$ since $x_{22}(t) \leq x_{21}(t)$ on $[0,t_2]$. Hence u_1 is time-optimal.

B. Suppose $x_{10} > 0$. We steer the response to hit the boundary in the fourth quadrant with $u = -1$, keep it on the boundary until it hits Γ_+ with $u = 0$, and with $u = +1$ we drive it to the origin. Also, in this case the above trajectory can be shown to be time-optimal in a similar way.

The proof of the theorem is thus complete. Once the control strategy is known, the (minimum) time taken by the optimal trajectory to reach the origin from any initial point $(x_{10},x_{20}) \varepsilon G \cap B^c$ can be easily shown to be:

$$T = \begin{cases} 1 - x_{20} + \frac{1}{2} x_{20}^2 - x_{10} & \text{if} \quad x_{10} < 0 \\ 1 + x_{20} - \frac{1}{2} x_{20}^2 + x_{10} & \text{if} \quad x_{10} > 0 \end{cases} \tag{14}$$

III. TIME-OPTIMAL PROBLEM WITH $\mu > 0$ in (4)

Consider $(x_{10},x_{20}) \varepsilon G \cap B^c$. Let $u(t)$ be an admissible control with domain of definition $[0,T]$.

Theorem 2:

Case 1. (a) $x_{10} < 0, x_{20} < 0$; (b) $x_{10} > 0, x_{20} > 0$.

For part (a), T is a minimum if and only if $[\tau_1,\tau_2] = [0,\tau]$ and $u(t) = 1 + \mu$ for almost all $t\varepsilon[\tau_1,\tau_2]$.

Case 2. (a) $x_{10} < 0, x_{20} = 0$; (b) $x_{10} > 0, x_{20} = 0$.

For part (a), T is a minimum if and only if $[\tau_1,\tau_2] = [0,\tau]$ and $u(t) = 1 + \mu$ for almost all $t\varepsilon[\tau_1,\tau_2]$ or $[\tau_1,\tau_2] = [T - \tau,T]$ and $u(t) = -1 - \mu$ for almost all $t\varepsilon[\tau_1,\tau_2]$.

Case 3. (a) $x_{10} < 0, x_{20} > 0$; (b) $x_{10} > 0, x_{20} < 0$.

For part (a), T is a minimum if any only if $[\tau_1,\tau_2] = [T - \tau,T]$ and $u(t) = -1 - \mu$ for almost all $t\varepsilon[\tau_1,\tau_2]$.

For part (b) in all cases everything applies if $1 + \mu$ is replaced by $-1 - \mu$ and vice versa.

Proof: It suffices to prove part (a) in all cases since part (b) can be proven similarly. Let us denote the trajectory obtained by utilizing the control given by Theorem 2 by $(x_{11}(t),x_{21}(t))$. Consider any other trajectory $(x_{12}(t),x_{22}(t))$ which starts at the same initial state (x_{10},x_{20}) at zero time and reaches the origin at time T'.

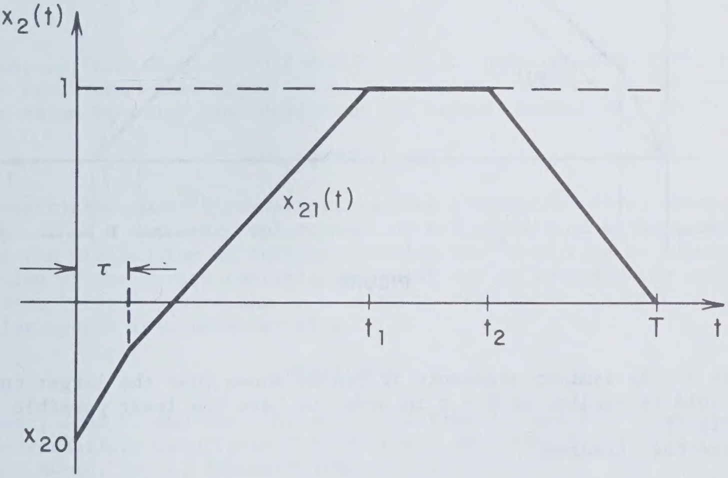


FIGURE 3

Case 1. (Fig. 3) The theorem states that the control should be $1 + \mu$ almost everywhere on $[0,\tau]$. If the larger control range is utilized before t_1,

$$\int_0^{t_2} x_{22}(t)\, dt \leq \int_0^{t_2} x_{21}(t)\, dt \tag{15}$$

and $x_{22}(t_2) \leq x_{21}(t_2)$.

In order for relation (6) to be satisfied, $x_{22}(t)$ can be zero only for all $t \geq T' \geq T$, the equality holding if and only if $1 + \mu$ is applied on $[0,\tau]$ almost everywhere. If the larger control range is utilized after t_2, in order that

$$\int_0^{T'} x_{22}(t)\,dt \quad \text{is equal to} \quad \int_0^{T} x_{21}(t)\,dt \quad ,$$

T' should be greater than T. Hence the minimum value of T is obtained when the larger control $1 + \mu$ is applied at zero time.

Case 2. If the larger control $1 + \mu$ is applied either at zero time or at time $T - \tau$, it can be observed that the area under the resulting curves, shown in Fig. 4, is the same. If the larger control is utilized at any other time, then $T < T'$. So T is the least possible value for the final time.

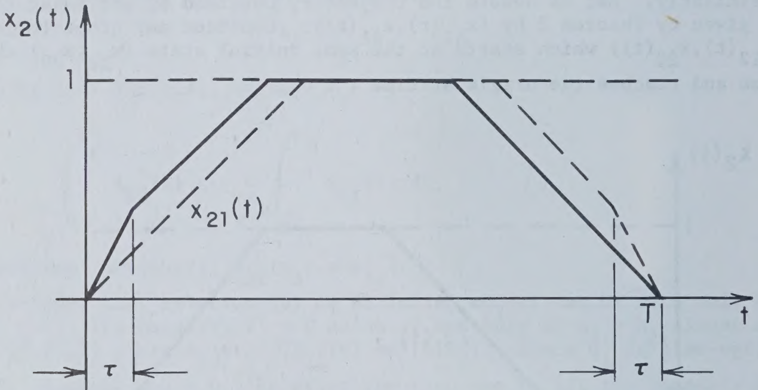


FIGURE 4

Case 3. By similar arguments it can be shown that the larger control $1 + \mu$ should be applied at $T - \tau$ in order to have the least possible value for T (Fig. 5).

Hence the theorem.

An Example

Let $\dot{x}_1 = x_2, \dot{x}_2 = u$, $|x_2| \leq 1$, $\tau = 0.5$, $\mu = 0.1$, $(x_{10},x_{20}) = (-4,-0.5)$.

The minimum time to reach the origin without utilizing the larger control range can be computed from equation (14).

$$T_1 = 5.625$$

In order to have the least possible value for the final time utilizing the larger control range, from Theorem 2, a control value of 1.1 should be applied for a period of time 0.5 starting at the initial time.

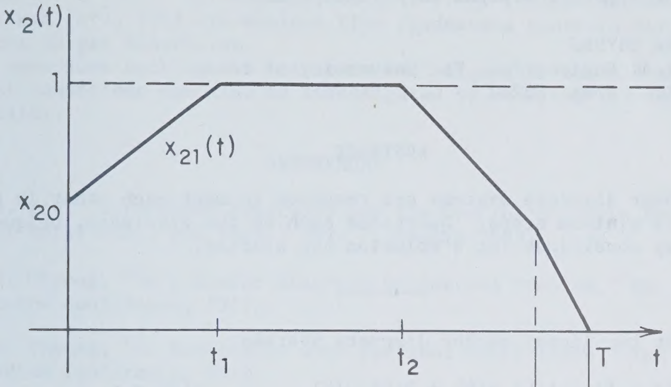


FIGURE 5

On [0,0.5] we have

$$x_2(t) = 1.1t - 0.5$$

$$x_1(t) = 1.1\frac{t^2}{2} - 0.5t - 4 \quad .$$

From this, $(x_1(0.5), x_2(0.5)) = (-4.1125, 0.05)$. From equation (14), the time taken to reach the origin from (-4.1125,0.05) is 5.06375. So the least possible value of final time utilizing the larger control is 5.56375.

IV. CONCLUSIONS

A restricted phase time-optimal control problem is solved using a new approach. Then a temporary enlargement of the control constraint set is made and its utilization in further lowering the final time is illustrated. Application of the larger control constraint set in lowering the cost functional of a certain system has to be done by taking the dynamics of that particular system into consideration.

REFERENCES

[1] Hauer, John F. and Hsu, Chih-Chi., "A Time-Optimal Problem with Recoverability Constraint," IEEE Trans. on Automatic Control, Vol. AC-14, No. 1, February 1969.

[2] Pontryagin, L. S., et al., The Mathematical Theory of Optimal Processes, Interscience Publishers, 1962.

[3] Fel'dbaum, A. A., Optimal Control Systems, Academic Press, 1965.

RENDEZVOUS OF DISCRETE SYSTEMS IN A TARGET AREA[*]

DONG HAK CHYUNG
Electrical Engineering, The University of Iowa
Iowa City, Iowa 52242

ABSTRACT

Two linear discrete systems are required to meet each other in a designated area in minimum steps. Questions such as the existence, uniqueness and necessary conditions for a solution are studied.

Consider two linear vector discrete systems

$$x(k+1) = A(k)\,x(k) + B(k)\,u(k) \quad , \qquad x(0) = x_0$$

$$y(k+1) = C(k)\,y(k) + D(k)\,v(k) \quad , \qquad y(0) = y_0$$

where x and y are n-dimensional state vectors, u and v are m-dimensional control vectors, and A, B, C and D are matrices with compatible dimensions. The control vectors u(k) and v(k) are in compact convex control restraint sets U and V, respectively. The sets U and V contain the zero vector. Let G be the target area, which is a closed convex set.

The problem is to find a control sequence pair {u(k)}, {v(k)} such that $x(k^*) = y(k^*)$ is in G with minimum $k^* \geq 0$. It is well known that the usual maximum principle does not apply to this kind of problem. Therefore we modify the problem as follows: For scalar constants $0 \leq a \leq 1$, $0 \leq b \leq 1$, define

$$U(a) = \{w \mid w = au,\ u \varepsilon U\}$$

$$V(b) = \{w \mid w = bv,\ v \varepsilon V\}$$

Then find a pair of control sequences $\{u^*(k)\}$ and $\{v^*(k)\}$ from $U(a^*)$ and $V(b^*)$, respectively, such that the corresponding responses $x^*(k^*) = y^*(k^*)$ are in G with minimum $k^* + (a^*/3) + (b^*/3)$. Note that this is also a solution to the original problem.

By studying the properties of the sets of attainability as in References [1-5], the following results can be obtained. Let $\{u^*(k)\}$, $\{v^*(k)\}$ be a solution pair. Then at least one of $\{u^*(k)\}$, $\{v^*(k)\}$ is a time optimal control to the target set G or there exist nontrivial solutions p(k), q(k) of the vector discrete equations

$$p(k) = p(k+1)\,A(k) \quad , \qquad q(k) = q(k+1)\,C(k)$$

for $0 \leq k \leq k^*$ with $p(k^*) = -q(k^*)$ such that

$$p(k)\,B(k)\,u^*(k) = \max_{u \varepsilon U(a^*)} p(k)\,B(k)\,u$$

$$q(k)\,D(k)\,v^*(k) = \max_{v \varepsilon V(b^*)} q(k)\,D(k)\,v$$

---

[*] This research was supported in part by NSF Grant GK-31628.

for all k, $0 \leq k \leq k^*$. Furthermore, if $x^*(k^*) = y^*(k^*)$ is not on the boundary of G then $\{u^*\}$, $\{v^*\}$ are minimum time rendezvous controls for the problem without target constraint.

The questions such as the existence and uniqueness as well as certain sufficient conditions can also be investigated by studying the sets of attainability.

REFERENCES

[1] E. B. Lee, "Recurrence Equations and the Control of Their Evolution," J. of Math. Anal. and Appl., Vol. 7, August 1963.

[2] D. H. Chyung, "On a Linear Discrete Rendezvous Problem," The Tenth Allerton Conference, 1972.

[3] D. H. Chyung, "On Rendezvous with Terminal Constraints," The Eleventh Allerton Conference, 1973.

[4] D. H. Chyung, "Time Optimal Rendezvous of Three Linear Systems," J. of Optimization Theory and Appl., Vol. 12, September 1973.

[5] D. H. Chyung, "On Nonunique Discrete Optimal Control," IEEE T-AC, Vol. AC-14, No. 3, June 1969.

OPTIMAL CIRCUIT DESIGN WITH A SENSITIVITY PENALTY

T. H. CHAU AND R. SAEKS
Texas Tech University
Lubbock, Texas 79409

I. INTRODUCTION

In the past few years computer-aided design has played an expanding role in circuit design growing in importance with the development of the high speed digital computer. Through the use of numberical techniques circuit design can be approached as an optimization problem wherein one attempts to choose circuit parameters so as to minimize a specified performance index. Such a performance index would ideally include a variety of attributes of the circuit under design including economic and realiability considerations as well as electrical performance. In practice, however, most existing circuit optimization algorithms use approximation error as the sole performance measure, economic and reliability considerations being considered in the a-priori choice of circuit configruation but not in the actual optimization.[1]

The purpose of the present paper is to formulate a circuit optimization procedure wherin one chooses component parameters to minimize a weighted sum of approximation error and circuit sensitivity. The results are predicated on some previous work of Ransom and Saeks[2,3,4] wherein separate sensitivity analysis and circuit optimization algorithms were developed, and like the previous work our algorithm for optimal circuit design with a sensitivity penalty yields a set of design equations expressed analytically in terms of the design specifications to be solved for the optimal component parameters.

We begin with a review of the component-connection model with which our design specifications will be formulated, and a review of the results of Ransom and Saeks on sensitivity analysis. Circuit optimization using approximation error as the only performance measure is reviewed in Section III, and a new set of design equations which are significently simplier than those of references 2 and 4 are derived for this optimation problem. In Section IV the design equations for the circuit optimization problem with a sensitivity penalty are derived, and the theory is illustrated with a filter design problem.

II. THE COMPONENT CONNECTION PROBLEM

By a component connection model for an electronic circuit we mean a circuit model wherein the components and connections are characterized by separate equations. In particular, we assume that the i^{th} component is characterized by an operator equation

$$b_i = z_i a_i \quad ; \quad i = 1,2, \ldots, n \tag{1}$$

mapping its input vector, a_i, into its output vector, b_i. For many purposes the abstract operator equation suffices though equation 1 may be rewritten in either the time or frequency domain as is convenient. Although in actual practice one normally works with the n separate component equations (for computational efficiency and core reduction) notationally they may be combined into the single equation

$$b = Za \tag{2}$$

where $b = col(b_i)$, $a = col(a_i)$ and $Z = diag(Z_i)$. Since the connections in most practical circuits are characterized by linear constraints (Kirchhoff laws, adders, scalers, etc.) we model our connections by the matrix equation

$$\begin{bmatrix} a \\ y \end{bmatrix} = \begin{bmatrix} L_{11} & L_{12} \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} b \\ u \end{bmatrix} \tag{3}$$

where y is the overall circuit output and u is the input vector for the overall circuit which are related by the overall circuit operator, S.

$$y = Su \tag{4}$$

Although non-standard and somewhat less intuitive than the traditional diagramatic and graphical connection models the four matrices of equation 3 exist and are readily computed for most practical circuits. Moreover, whatever is lost in intuition over the graphical and diagramatic connection models is more than made up by the ease with which our algebraic connection model may be manipulated, both analytically and computationally.

Upon comparing equations 2 and 3 with 4 one obtains the equation

$$S = L_{22} + L_{21}(1 - ZL_{11})^{-1} ZL_{12} = f(Z) \tag{5}$$

realting the overall circuit operator to the composite component operator. Assuming that the connections are fixed, as it the case in our optimization problem, equation 5 may be interpreted as a system valued, S, function of a component valued, Z, variable. Since this function is entirely determined by the fixed connection matrices it has been termed the connection function[2]. Of course if we let the operators Z and S be characterized in the frequency domain we obtain

$$S(\omega) = f(Z(\omega)) \tag{6}$$

since the connections are independent of frequency. In many practical circuits one can write the component matrix, $Z(\omega)$, in the form

$$Z(\omega) = E(\omega)CF(\omega) \tag{7}$$

where $E(\omega)$ and $F(\omega)$ are frequency dependent matrices determined by the componen-types but not their actual values and C is a frequency independent matrix of component values (i.e. for an inductor the $j\omega$ goes in E or F and L goes in the C matrix). Such a point of view is quite reasonable for our optimization problem where we may assume that the component types have been pre-specified along with the circuit topology and our only design latitude is the choice of the component parameter matrix, C. We then combine the known E and F matrices into our connection function and characterize our circuit by

$$S(\omega) = f(E(\omega)CF(\omega)) = f_{\omega}(C) \tag{8}$$

which form is ideally suited for our optimization problem since the function, f_{ω}, is entirely determined by the circuit design specifications whereas C is our design variable.

Finally, we consider the special form which the connection function takes on when, as is often the case, the composite component matrix, $Z(\omega)$, is diagonal[4]. In this case we may lump E and F together into a single matrix and deal with

$$Y(\omega) = Z(\omega)^{-1} = B(\omega)D \qquad (9)$$

where $D = C^{-1}$ is also diagonal. We then obtain the simplified expressions

$$S = L_{22} + L_{21}\,(Y - L_{11})^{-1}\,L_{12} = \tilde{f}(Y) \qquad (10)$$

and

$$S(\omega) = f(Y(\omega)) = f(B(\omega)D) = f_{\omega}\,(D) \qquad (11)$$

From the point of view of our applications to computer-aided design and senstivity analysis the key attribute of the connection function discription of a circuit is that one can differentiate f_ω(or $\tilde{f}_\omega$) analytically, thereby obtaining both an explicit formula for the circuit's sensitivity and an explicit expression for the Kuhn-Tucker equations of an optimization problem when the performance index is expressed in terms of the connection function. In particular, if r is an arbitrary scalar parameter

$$\frac{dS}{dr} = \left[L_{21}(1 - ZL_{11})^{-1}\right]\left[\frac{dZ}{dr}\right]\left[L_{11}(1 - ZL_{11})^{-1}\,Z + 1\right]\,L_{21} \qquad (12)$$

and

$$\frac{dS}{dr} = \left[L_{21}(Y - L_{11})^{-1}\right]\left[\frac{dY}{dr}\right]\left[(Y - L_{11})^{-1}\right]L_{12} \qquad (13)$$

for the general case and the diagonal case respectively. Now, if the sensitivity of a parameter, y, to a parameter, r, is defined as the usual per unit derivative

$$\mathcal{S}_r^y = \frac{dy}{dr}\,\frac{r}{y} \qquad (14)$$

and the sensitivity of a vector Y to a vector of parameters R is taken to be the matrix

$$\mathcal{S}_R^Y = \left(\mathcal{S}_{r_j}^{y_i}\right) \qquad (15)$$

then we may characterize the sensitivity of S(ω) with respect to R by the matrix

$$\mathcal{S}\frac{S}{R} = \hat{S}^{-1}\left(\left[\left[L_{11}(1-ZL_{11})^{-1}\,Z + 1\right]\,L_{12}\right]^T \otimes \left[L_{21}(1-ZL_{11})^{-1}\right]\,\hat{Z}\right)\mathcal{S}\frac{Z}{R} \qquad (16)$$

in the general case and

$$\mathcal{S}\,\frac{S}{R} = \left(\hat{S}^{-1}\,\left[(Y-L_{11})^{-1}\,L_{12}\,\right]^T \odot \left[\,L_{21}(Y-L_{11})^{-1}\right]\,Y\right)\mathcal{S}_R^{\overline{Y}} \qquad (17)$$

in the diagonal case. Here $\underline{S}$ denotes the vector obtained by taking the columns of S and stacking them on top of one another, $\hat{S}$ denotes the diagonal matrix obtained by taking the columns of S and stacking them along the diagonal of $\hat{S}$. $\underline{Z}$ and $\hat{Z}$ are similarly defined whereas $\overline{Y}$ is the column vector obtained from the diagonal entries of Y, and $\otimes$ and $\odot$ denote the Kronecker cross and dot product respectively[6] Although our notation is somewhat nonstandard equations 16 and 17 yield a straightforward tabulation of the sensitivities of all entries in S with respect to all parameters in R.

III. THE OPTIMAL DESIGN PROBLEM

Given the above notation we may now readily formulate our optimal circuit design problem. For this purpose we assume that we are given an ideal desired frequency response; $T(\omega_i)$, $i = 1,2, \ldots, k$; at k frequencies; and the circuit topology and component types have been specified (i.e. $\tilde{f}_\omega$ is given). In the case where our performance index is dependent only on approximation error we then desire to choose the component parameter matrix, D, so as to minimize

$$J(D) = \sum_{i=1}^{k} || T(\omega_i) - f_{\omega_i}(D) ||_M^2 \qquad (18)$$

$$D \geq 0$$

Here the inequality constraints corresponds to the requirement that the component values be positive and J(D) is a measure of the approximation error at the k sample frequencies. Finally, our matrix norm in equation 18 is taken to be

$$||A||_M^2 = \sum_{i,j} |a_{ij}|^2 = ||\underline{A}||^2 = \mathrm{tr}(A^\dagger A) \qquad (19)$$

Equations 18 are precisely in the form of a non-linear programing problem and hence the solution must satisfy the Kuhn-Tucker Equations

$$\begin{gathered} \nabla_D J(D) + \lambda = 0 \\ \lambda_j d_j = 0 \; : \; j = 1,2, \ldots, n \end{gathered} \qquad (20)$$

where $\lambda = \mathrm{col}(\lambda_i)$ is a Lagrange multiplier and d_i is the i^{th} component parameter (i.e. $D = \mathrm{diag}(d_i)$). Unlike most optimization problems it was found in references 2 and 4 that these equations could be expressed analytically in terms of the design specifications (the $T(\omega_i)$, $B(\omega_i)$ and L_{ij} matrices). Unfortunately, the resultant expression was so complex as to be of little value. If, however, we work with the form of the norm written in terms of the trace operator rather than the form written in terms of $\underline{A}$ as was done in those references and exploit the fact that the trace operation commutes with differentiation the Kuhn-Tucker equations reduce to the following set of 2n non-linear algebraic equations in 2n unknowns

$$\sum_{i=1}^{k} -2\mathrm{Re}\ \mathrm{tr}\left[T(\omega_i)-f_{\omega_i}(D)\right]^\dagger \left[L_{21}(B(\omega_i)D-L_{ii})^{-1}B_j(\omega_i)(B(\omega_i)D-L_{11})^{-1}L_{12}\right] + \lambda_j = 0$$

and

$$\lambda_j d_j = 0 \; , \; j = 1, 2, \ldots, n \qquad (21)$$

Here $B_j(\omega_i)$ is a matrix all of whose entries are zero except for the j-j entry which coincides with the j-j entry of $B(\omega_i)$, and all other terms are as previously defined. Although equations 21 are certainly not simple they do represent a set of design equations for our optimization problem which may be computer solved with significantly more efficiency than the traditional "hunt and search" circuit optimization algorithms. Moreover, they represent a set of circuit design equations expressed analytically in terms of the problem specifications, we believe for the first time in this degree of generality.

IV. OPTIMAL DESIGN WITH A SENSITIVITY PENALTY

Finally, we may choose to combine the above optimization procedure with our sensitivity formula minimizing

$$K(D) = \sum_{i=1}^{k} \|T(\omega_i) - f_{\omega_i}(D)\|_M^2 + \varepsilon_i \|\mathcal{S}\frac{S}{D}(\omega_i)\|_M^2 \tag{22}$$

Subject to the constraints

$$D \geq 0 \tag{23}$$

As before we must solve the Kuhn-Tucker Equations

$$\begin{aligned} \nabla_D K(D) + \lambda &= 0 \\ \lambda_j d_j = 0 \quad j &= 1,2, \dots, n \end{aligned} \tag{24}$$

which once again may be expressed analytically in terms of the design specifications. These equations take the form

$$\sum_{i=1}^{k} 2\mathrm{Re}\ \mathrm{tr}\ \left[T(\omega_i) - f_{\omega_i}(D)\right]^{+}\left[\frac{\partial}{\partial d_j} f_{\omega_i}(D)\right] + \varepsilon_i \left[\mathcal{S}\frac{S}{D}(\omega_i)\right]^{+}\left[\frac{\partial}{\partial d_j}\mathcal{S}\frac{S}{D}(\omega_i)\right] + \lambda_j = 0$$

and

$$\lambda_j d_j = 0 \quad j = 1,2, \dots, n \tag{25}$$

As in the previous case the derivatives can be evalued explicitally in terms of the $T(\omega_i)$, $B(\omega_i)$ and L_{ij} matrices. These rather complex expressions appear in reference 5 and will not be repeated here. Suffice it to say that they yield a set of design equations for solving our optimal design problem via a digital computer. As before, an explicit set of design equations are obtained.

As an illustration of the theory two designs for a low pass filter using the configuration of Figure 1 were developed. In the first design no sensitivity penalty was used and in the second design the sensitivity term was weighted at 1/4 the approximation error term for frequencies in the passband and zero outside the passband. The resultant transfer function and sensitivity magnitude curves are shown in Figure 2. The second design shows a clear sensitivity improvement in the passband at a cost in approximation error which in this case manifests itself as an upward migration of the cut-off frequency.

V. REFERENCES

1. Szentirmai, G., Computer-Aided Circuit Design, New York, IEEE Press, 1973
2. Ransom, M.N., and R. Saeks, "The Connection Function - Theory and Application", Circuit Theory and Applications, Vol. 2, (to appear).
3. Ransom, M.N., and R. Saeks, "A Functional Approach to Large-Scale Systems", Proc. of the 10th Allerton Conference on Circuits and Systems, University of Illinois, pp. 48-55, (1972).
4. Ransom, M.N., Ph.D. Thesis, University of Notre Dame, 1973.
5. Chau, T. H., M.S. Thesis, Texas Tech University, 1974.
6. Rao, C.R., and S. K. Mitra, Generalized Inverse of Matrices and Its Applications, New York, J. Wiley and Sons, 1971.

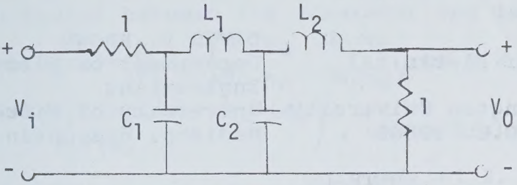


Figure 1. Filter Configuration

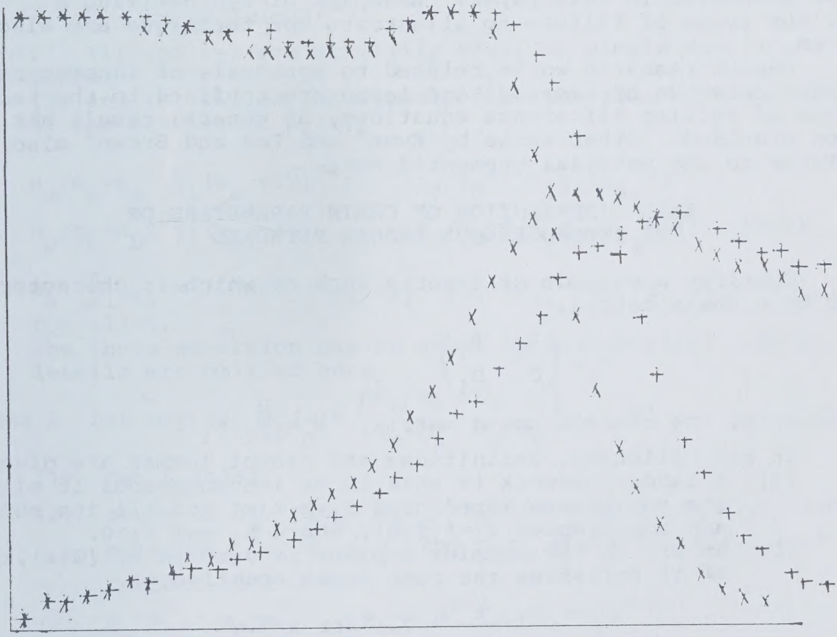


Figure 2. Transfer Function and Sensitivity Magnitude curves for Filter designs with (+) and without (X) sensitivity penalty.

DECOMPOSITION THEOREM - SYNTHESIS OF INHOMOGENEOUS LADDER NETWORKS

TING N. LEE
Department of Electrical Engineering
George Washington University
Washington, D.C. 20006

DAVID P. BROWN
Department of Electrical Engineering
University of Wisconsin
Madison, Wisconsin 53706

ABSTRACT

The decomposition property of the chain parameter matrix of the inhomogeneous ladder networks is investigated in this paper. It is found that the zeros of the chain parameters of the corresponding networks can be expressed by a simple root locus equation. This expression can then be used for synthesis purpose. Starting with a 2 by 2 matrix, the Iff. conditions are found (mainly due to the zero distribution patterns of the entries of the matrix), such that this matrix is decomposed into a product of simple 2 by 2 matrices each of which represents a simple ladder. The uniqueness of the decomposition is also discussed in this paper. Examples of synthesizing a various types of filters to illustrate the technique are also given.

Resent research works related to synthesis of inhomogeneous ladder networks by Lampard[1] and Levan are confined to the technique of solving difference equations, no general result has been provided. Other works by Hunt[2] and Lee and Brown[3] also relates to the material presented here.

ZERO DISTRIBUTION OF CHAIN PARAMETERS OF THE INHOMOGENEOUS LADDER NETWORKS

Consider a cascade of 2-ports each of which is characterized by a chain matrix,

$$K_i = \begin{pmatrix} A_i & B_i \\ C_i & D_i \end{pmatrix} .$$

Therefore, the overall chain matrix, $S_n = \prod_{i=1}^{n} K_i$.

In the following, definitions and useful lemmas are given.

(1) A ladder network is said to be inhomogeneous if all its series arm impedances $Z_i = k_i Z(s)$ and all its shunt arm admittances $Y_i = \ell_i Y(s)$, where k_i and $\ell_i > 0$.

(2) An arc in the complex s-plane is denoted by $[G(s),k]$, if it satisfies the root locus equation of

$$1 + \frac{k}{G(s)} = 0, \quad \text{for } k \geq 0.$$

(3) The zeros of two polynomials $A(s)$ and $B(s)$ are said to be alternative with respect to $[G(s),k]$, if the zeros of $A(s)$ are described by equation $1 + \frac{ka_i}{G(s)} = 0$, and the zeros of $B(s)$ are described by equation $1 + \frac{k_{b_i}}{G(s)} = 0$, such that $0 \leq k_{a_i} < k_{b_i}$ or $0 \leq k_{b_i} < k_{a_i}$, for $i = 1, 2, \ldots$

It is assumed without further mentioning throughout this paper that each of the chain matrix entries does not consist of a common factor between its numerator and denominator.

Lemma 1 Let $S_n(s)=\prod_{i=1}^{n} K_i=\begin{pmatrix} n_a/m_a & n_b/m_b \\ n_c/m_c & n_d/m_d \end{pmatrix}$ be a 2 by 2 matrix. If

$$K_i=\begin{pmatrix} 1+k_i\ell_i Z(s)Y(s) & k_i Z(s) \\ \ell_i Y(s) & 1 \end{pmatrix}, \text{ where } Z(s)=n_1/m_1,\ Y(s)=n_2/m_2 \text{ and } i=1,2,\ldots,n,$$

then (1) $\mathrm{Det}.S_n(s)=1$, $n_a/m_a|_{s=z_{n(s)}}=1$, $z_{n(s)}$ are the zeros of the polynomial $n(s)=n_1n_2$,

(2) zeros of n_a, n_b, n_c and n_d lie on $[Z(s)Y(s),k]$,

(3) zeros of n_a and n_2n_b, n_a and n_1n_c, n_2n_b and n_d, n_1n_c and n_d are alternative with respect to $[Z(s)Y(s),k]$,

(4) $m_a=[m_1m_2]^n$, $m_d=[m_1m_2]^{n-1}$, $m_b=m_1[m_1m_2]^{n-1}$, and $m_c=m_2[m_1m_2]^{n-1}$.

Proof: (1) and (4) are trivially true, by simple matrix algebra.

(2) and (3) follow immediately from the general expressions of the entries of $S_n(s)$,

$$\text{(A)}\quad \begin{aligned} n_a/m_a&=k_a\prod_{i=1}^{n}(z_{a_i}+ZY)\ ; & n_d/m_d&=k_d\prod_{i=1}^{n-1}(z_{d_i}+ZY)\ ; \\ n_b/m_b&=k_bZ\prod_{i=1}^{n-1}(z_{b_i}+ZY)\ ; & n_c/m_c&=k_cY\prod_{i=1}^{n-1}(z_{c_i}+ZY), \end{aligned}$$ where

$k_a\prod_{i=1}^{n} z_{a_i}=1$, and $z_{a_i}<z_{b_i}$, $z_{a_i}<z_{c_i}$, $z_{b_i}<z_{d_i}$, $z_{c_i}<z_{d_i}$, for all i.

The above assertion can be shown by mathematical induction, details are omitted here.

Lemma 2 Let matrix $S_n(s)=\begin{pmatrix} n_a/m_a & n_b/m_b \\ n_c/m_c & n_d/m_d \end{pmatrix}$. If

(1) $\mathrm{Det}.S_n(s)=1$,

(2) zeros of n_a, n_2n_b, n_1n_c, n_d lie on $[n(s)/m(s),k]$ and moreover, zeros of n_a and n_1n_c are alternative with respect to the same arc,

(3) $m_a=m^n$, $m_d=m^{n-1}$, $m_b=m_1m^{n-1}$, $m_c=m_2m^{n-1}$,

(4) a. for degree of polynomial n(s)>degree of polynomial m(s):

$\mathrm{degree}(n_a)=\mathrm{degree}(n(s)^n)$, $\mathrm{degree}(n_d)=\mathrm{degree}(n(s)^{n-1})$, $\mathrm{degree}(n_b)=\mathrm{degree}(n_1n(s)^{n-1})$, $\mathrm{degree}(n_c)=\mathrm{degree}(n_2n(s)^{n-1})$

where $n(s)=n_1n_2$ and $m(s)=m_1m_2$,

b. for degree of $m(s)$ degree of $n(s)$:

$\text{degree}(n_a)=\text{degree}(m(s)^n)$, $\text{degree}(n_d)=\text{degree}(m(s)^{n-1})$,

$\text{degree}(n_b)=\text{degree}(n_1m(s)^{n-1})$,

$\text{degree}(n_c)=\text{degree}(n_2m(s)^{n-1})$,

then zeros of n_a and n_2n_b, n_2n_b and n_d, n_1n_c and n_d are alternative with respect to $[n(s)/m(s),k]$.

This lemma follows from the property of $\text{Det}.S_n(s)=1$. The proof is similar to the proof given in Lee and Brown[3] and therefore details are omitted.

Lemma 3 Let matrix $S_n(s)=\begin{pmatrix} n_a/m_a & n_b/m_b \\ n_c/m_c & n_d/m_d \end{pmatrix}$. If

(1) $\text{Det}.S_n(s)=1$,

(2) zeros of n_a, n_2n_b, n_1n_c, n_d lie on $[n(s)/m(s),k]$ and moreover, zeros of n_a and n_2n_b are alternative with respect to the same arc,

(3) $m_a=m^{n-1}$, $m_d=m^{n-1}$, $m_b=m_1m^{n-1}$, $m_c=m_2m^{n-1}$,

(4) a. for degree of $n(s)>$degree of $m(s)$:

$\text{degree}(n_a)=\text{degree}(n(s)^{n-1})$, $\text{degree}(n_d)=\text{degree}(n(s)^{n-1})$

$\text{degree}(n_b)=\text{degree}(n_1n(s)^{n-1})$,

$\text{degree}(n_c)=\text{degree}(n_2n(s)^{n-1})$,

b. for degree of $m(s)>$degree of $n(s)$:

$\text{degree}(n_a)=\text{degree}(m(s)^{n-1})=\text{degree}(n_d)$,

$\text{degree}(n_b)=\text{degree}(n_1m(s)^{n-1})$,

$\text{degree}(n_c)=\text{degree}(n_2m(s)^{n-1})$,

where $n(s)=n_1n_2$ and $m(s)=m_1m_2$

Then zeros of n_a and n_1n_c, n_2n_b and n_d, n_1n_c and n_d are alternative with respect to $[n(s)/m(s),k]$.

DECOMPOSITION THEOREMS

Applying the above lemmas, the following Decomposition Theorems- DFR (Decomposition From The Right), and DFL (Decomposition From The Left), are proven.

Theorem 1 (DFL) Let matrix $S_n(s)=\begin{pmatrix} n_a/m_a & n_b/m_b \\ n_c/m_c & n_d/m_d \end{pmatrix}$.

$S_n(s)=k_nK_{n-1}..K_1$, where $K_i=\begin{pmatrix} 1+k_i\ell_iZY & k_iZ \\ \ell_iY & 1 \end{pmatrix}$, k_i, $\ell_i>0$,

and $Z=n_1/m_1$, $Y=n_2/m_2$, iff.

(1) $\text{Det}.S_n(s)=1$, $n_a/m_a\big|_{z_{ZY}}=1$, where z_{ZY} are the zeros of ZY,

(2) zeros of n_a, n_2n_b, n_1n_c, n_d lie on $[ZY, k]$ and moreover,

zeros of n_a and $n_1 n_c$ are alternative with respect to [ZY, k], starting with the zeros of $n_1 n_c$ at k=0,

(3) $m_a = m^n$, $m_d = m^{n-1}$, $m_b = m_1 m^{n-1}$, $m_c = m_2 m^{n-1}$,

(4) a. for degree(n(s))>degree(m(s)):

$\text{degree}(n_a)=\text{degree}(n(s)^n)$, $\text{degree}(n_d)=\text{degree}(n(s)^{n-1})$,

$\text{degree}(n_b)=\text{degree}(n_1 n(s)^{n-1})$,

$\text{degree}(n_c)=\text{degree}(n_2 n(s)^{n-1})$,

b. for degree(m(s))>degree(n(s))

$\text{degree}(n_a)=\text{degree}(m(s)^n)$, $\text{degree}(n_d)=\text{degree}(m(s)^{n-1})$,

$\text{degree}(n_b)=\text{degree}(n_1 m(s)^{n-1})$,

$\text{degree}(n_c)=\text{degree}(n_2 m(s)^{n-1})$,

where $m(s)=m_1 m_2$, $n(s)=n_1 n_2$.

Proof: Necessity: This part of the proof follows immediately from Lemma 1.

Sufficiency: It follows from Lemma 2, zeros of $n_1 n_c$ and n_d are alternative with respect to [ZY, k], starting with the zeros of $n_1 n_c$ at k=0.

Since (2), (3) and (4) imply that n_a/m_a, n_b/m_b, n_c/m_c and n_d/m_d can be factored into the expression as shown in (A) of Lemma 1, hence

$$n_a/m_a = a_n (ZY)^n + a_{n-1}(ZY)^{n-1} + \ldots + 1 \ ,$$
$$n_c/m_c = Y[c_{n-1}(ZY)^{n-1} + c_{n-2}(ZY)^{n-2} + \ldots + c_0] .$$

(B) By letting $k_n = a_n / c_{n-1}$ and $\ell_n = c_{n-1}^2 / a_{n-1}c_{n-1} - a_n c_{n-2}$, it is easily seen that

$$S_{n-1}(s) = \begin{pmatrix} n'_a/m'_a & n'_b/m'_b \\ n'_c/m'_c & n'_d/m'_d \end{pmatrix} = \begin{pmatrix} 1+k_n \ell_n ZY & k_n Z \\ \ell_n Y & 1 \end{pmatrix}^{-1} \begin{pmatrix} n_a/m_a & n_b/m_b \\ n_c/m_c & n_d/m_d \end{pmatrix} ,$$

where n'_a/m'_a, n'_b/m'_b, n'_c/m'_c and n'_d/m'_d of matrix $S_{n-1}(s)$ satisfy condition (1) through (4) again except each of the polynomials is one degree less than the corresponding original polynomial, by Lemma 2 and 3.

This process is then continued until n=1 which is a trivial case. This completes the proof of the sufficiency part of the proof. Q.E.D.

Theorem 2 (DFR) Let matrix $S_n(s) = \begin{pmatrix} n_a/m_a & n_b/m_b \\ n_c/m_c & n_d/m_d \end{pmatrix}$.

$S_n(s) = K_1 K_2 \ldots K_n$, where $K_i = \begin{pmatrix} 1+k_i \ell_i ZY & k_i Z \\ \ell_i Y & 1 \end{pmatrix}$, $k_i, \ell_i > 0$ and $Z = n_1/m_1$, $Y = n_2/m_2$, iff.

(1) Det.$S_n(s)=1$, $n_a/m_a|_{z_{ZY}}=1$, where z_{ZY} are the zeros of ZY,

(2) zeros of n_a, n_2n_b, n_1n_c, n_d lie on [ZY, k] and moreover, zeros of n_a and n_2n_b are alternative with respect to [ZY, k] starting with the zeros of n_2n_b at k=0,

(3) $m_a=m^n$, $m_d=m^{n-1}$, $m_b=m_1m^{n-1}$, $m_d=m_2m^{n-1}$,

(4) same as (4) of Theorem 1.

The proof of this theorem is similar to the proof of the Theorem 1's and therefore will not be given here.

It is worthwhile to point out at this juncture to the reader that the values of k_i and ℓ_i for all i, obtained from (B) of Theorem 1, are indeed coincident to the values obtained from the method of continued fraction expension. In what follows, a uniqueness theorem is developed.

Theorem 3 (Uniqueness) It conditions of Theorem 1 (or Theorem 2) are true, then the pair of entries of $S_n(s)$, n_a/m_a and n_c/m_c (or n_b/m_b) uniquely determine the other pair of entries n_d/m_d and n_b/m_b (or n_c/m_c). In fact, they can be computed as given in the following matrix form,

$$\text{(C)}\quad \begin{bmatrix} a_n & 0 & 0 \ldots 0 & -c_{n-1} & 0 & \ldots \quad 0 \\ a_{n-1} & a_n & 0 \ldots 0 & -c_{n-2} & -c_{n-1} & \ldots 0 \\ a_{n-2} & a_{n-1} & a_n \,..\, 0 & -c_{n-3} & -c_{n-2} & \ldots 0 \\ .. & .. & \ldots..0 & .. & .. & .0 \\ & & \ldots & & \ldots & \\ a_0 & a_1 & a_2 \,..\, 0 & -c_1 & -c_2 & ..\; 0 \\ 0 & a_0 & a_1 \,..\, a_n & 0 & -c_1 & ..-c_{n-1} \\ 0 & 0 & a_0 \quad a_{n-1} & 0 & 0 & \ldots -c_{n-2} \\ .. & .. & 0 \quad \ldots & \ldots & \ldots & \ldots \\ & & \quad\ \ldots & & \ldots & -c_1 \\ 0 & 0 & 0 \,..\, a_0 & 0 & 0 & \ldots\; 0 \end{bmatrix} \begin{bmatrix} d_{n-1} \\ d_{n-2} \\ d_{n-3} \\ .. \\ .. \\ d_0 \\ b_{n-1} \\ b_{n-2} \\ .. \\ .. \\ b_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ . \\ . \\ . \\ . \\ 0 \\ 0 \\ . \\ 1 \end{bmatrix}$$

where a_i, b_i, c_i and d_i are the coefficients of the corresponding polynomials of n_a, n_b, n_c and n_d respectively.

Or

$$
\begin{bmatrix}
a_n & 0 & 0 & \cdots 0 & -b_{n-1} & 0 & \cdots & 0 \\
a_{n-1} & a_n & 0 & \cdots 0 & -b_{n-2} & -b_{n-1} & \cdots & 0 \\
a_{n-2} & a_{n-1} & a_n & \cdots 0 & -b_{n-3} & -b_{n-2} & \cdots & 0 \\
\cdot & \cdot & \cdot & \cdot\cdot\ 0 & \cdot & \cdot & \cdot\cdot & 0 \\
 & \cdots & & & & \cdots & & \\
a_0 & a_1 & a_2 & \cdot\cdot\ 0 & -b_1 & -b_2 & \cdot\cdot & 0 \\
0 & a_0 & a_1 & \cdot\cdot\ 0 & 0 & -b_1 & \cdot\cdot & -b_{n-1} \\
0 & 0 & a_0 & \cdot\cdot\ 0 & 0 & 0 & \cdot\cdot & \\
 & & & \cdots & & & & -b_1 \\
0 & 0 & 0 & \cdot\cdot\ a_0 & 0 & 0 & \cdot\cdot & 0
\end{bmatrix}
\begin{bmatrix}
d_{n-1} \\ d_{n-2} \\ d_{n-3} \\ \cdot \\ \cdot \\ d_0 \\ c_{n-1} \\ \cdot \\ \cdot \\ c_1
\end{bmatrix}
=
\begin{bmatrix}
0 \\ 0 \\ 0 \\ \cdot \\ \cdot \\ 0 \\ \cdot \\ \cdot \\ 0 \\ 1
\end{bmatrix}
\tag{D}
$$

The proof of this theorem follows immediately from the continued fraction expension and some tedious algebraic manipulations. Details are omitted.

APPLICATIONS- FILTER DESIGN

The Decomposition Theorem as well as the uniqueness property of the specified matrix $S_n(s)$ are used to synthesize classes of filter below. In what follows synthesis of Butterworth filter with 1 Ω termination is derived. Results of Bessel and Chebyshev filters are also provided.

Readers are asked to bear in mind that for a ladder low-pass filter, $Z(s)=Y(s)=s$. It is easily seen that the entries of $S_n(s)$ are polynomials, which imply $m_a=m_b=m_c=m_d=1$ and zeros of n_a and n_b, n_a and n_c, n_b and n_d, n_c and n_d are alternative with respect to $[s^2, k]$ which is the imaginary axis. Furthermore, n_a and n_d are even polynomials, n_c and n_b are odd polynomials.

a. 4th order Butterworth filter design:

Given: $|V_2/V_1|^2 = G(s)G(-s)|_{s=j\omega}$, where

$G(s)= 1/s^4+2.613s^3+3.414s^2+2.613s+1$.

Steps:

1) Since $\begin{pmatrix} V_1 \\ I_1 \end{pmatrix} = \begin{pmatrix} n_a & n_b \\ n_c & n_d \end{pmatrix} \begin{pmatrix} V_2 \\ I_2 \end{pmatrix}$,

hence, $V_1=n_aV_2+n_bI_2$, but $V_2^2/I_2=1$,

$V_2/V_1=1/n_a+n_b$.

2) Since n_a=Even polynomial and n_b=Odd polynomial,

hence, $n_a=s^4+3.414s^2+1$ and $n_b=2.613s^3+2.613s$.

3) It follows from (D) of Theorem 3,

$n_c=0.656s^3+1.962s$

$n_d=1.715s^2+1$.

4) From Theorem 2, it yields,

$\ell_2=0.380$, $\ell_1=1.580$ and

$k_2=1.085$, $k_1=1.582$.

5) Therefore network of Fig. 1 gives the required characteristics with

$$S_n(s)=\begin{pmatrix} s^4+3.414s^2+1 & 2.613s^3+2.613s \\ 0.656s^3+1.962s & 1.715s^2+1 \end{pmatrix}$$

b. 4th order Bessel filter design:

It follows from given $G(s)=1/105s^4+105s^3+45s^2+10s+1$,

$n_a=105s^4+45s^2+1$, $n_b=105s^3+10s$.

Therefore,

$$S_n(s)=\begin{pmatrix} n_a & n_b \\ 15s^3+6s & 15s+1 \end{pmatrix}$$

Hence, $\ell_2=1$, $\ell_1=5$ and $k_2=3$, $k_1=7$.

The network is shown in Fig. 2

c. 4th order Chebyshev filter design:

It follows from given $G(s)=1/0.379s^4+1.025s^3+1.716s^2+1.197+1$ which is for 1/2 db ripple in the passband,

$n_a=0.379s^4+1.716s^2+1$ and $n_b=1.025s^3+1.197s$.

Therefore,

$$S_n=\begin{pmatrix} n_a & n_b \\ \begin{matrix}0.970s^3+ \\ +3.620s\end{matrix} & 2.620s+1 \end{pmatrix}$$

Hence, $\ell_2=0.37$, $\ell_1=3.25$ and $k_2=0.805$, $k_1=0.392$.

The network is shown in Fig. 3.

ACKNOWLEDGEMENT: Authors would like to thank Dr. Leonard Magid of N.S.F. for the encouragement to tackle the problem. They also wish to thank the National Science Foundation for the partial supporting this work under Grant GK-42118.

REFERENCES

1. Lampard, D.G.,"Inhomogeneous Ladder Networks Part 1" The International J. of Circuit Theory and Applications, John Wiley and Sons, Sug. 1973.
2. Hunt, M.S.,"Synthesis of a Cascaded Networks by Transfer Matrix Factorization" IRE(London), pp. 203-205, March 1967.
3. Lee, T.N. and Brown, D.P.,"Synthesis of Two-Element-Kind Cascade Networks" Proc. 10th Allerton Conf. on Circuit and System Theory, pp. 692-699, 1972.

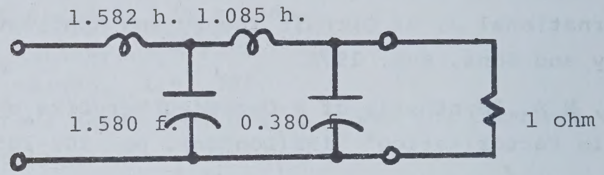

1.582 h.
1.085 h.
1.580 f.
0.380 f.
1 ohm

Fig. 1 4th order Butterworth Filter

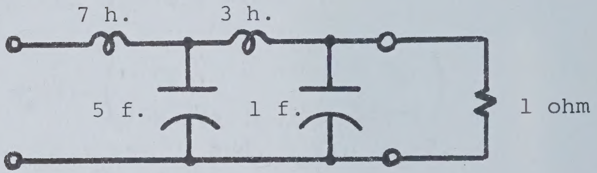

7 h.
3 h.
5 f.
1 f.
1 ohm

Fig. 2 4th orer Bessel Filter

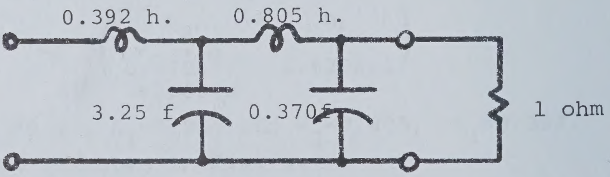

0.392 h.
0.805 h.
3.25 f
0.370f
1 ohm

Fig. 3 4th order Chebyshev Filter

SOME PROPERTIES AND APPLICATIONS OF MATRIX CONTINUED FRACTION

L. S. SHIEH AND F. F. GAUDIANO
Department of Electrical Engineering
University of Houston
Houston, Texas

ABSTRACT

The formulation of matrix continued fraction of three Cauer matrix forms and their dynamic state equations for linear time-invariant multivariable circuits and systems is derived. Some properties and applications of matrix continued fraction are discussed.

INTRODUCTION

The formulations and applications of continued fraction of the first and the second Cauer forms to single variable circuits and systems are well known.[1,2,3] The continued fraction of the second Cauer form[1] has been successfully extended to the matrix continued fraction for a multivariable system by Chen[4] and Shieh[5] etal. The structure of matrix continued fraction of the second Cauer matrix form has been defined in a way quite analogous to that of a single variable system.

Our purpose is to extend the formulation of the first[1] and the third[6] Cauer forms to the Cauer matrix forms for multivariable systems and to investigate the dynamic structures and properties of all three Cauer matrix forms. An algebraic method is then established for their applications.

FORMULATION OF CAUER MATRIX FORMS

A transfer-function matrix of a linear time-invariant multivariable system can be formulated as follows.

$$[T(s)] = [A_{21}+A_{22}s+\ldots+A_{2,n}s^{n-1}][A_{11}+A_{12}s+\ldots+A_{1,n+1}s^{n}]^{-1} \qquad (1)$$

where each $A_{i,j}$ is a real and constant m by m matrix, and $A_{1j} = a_j[I]$, $j = 1,\ldots,n+1$ where each a_j is a coefficient of the common-denominator polynomial or $\Delta(s) = \sum_{j=1}^{n+1} a_j s^{j-1}$ and $[I]$ is an identity matrix. The structure of the matrix continued fraction of three Cauer matrix forms for multivariable systems is similar to that of the Cauer forms[6] for single variable systems. The three Cauer matrix forms are expressed as follows:

1) The first Cauer matrix form is

$$[T(s)] = [H_1's+[H_2'+[H_3's+[H_4'+[\ldots]^{-1}]^{-1}]^{-1}]^{-1}]^{-1} \qquad (2)$$

2) The second Cauer matrix form is

$$[T(s)] = [H_1+[H_2\frac{1}{s}+[H_3+[H_4\frac{1}{s}+[\ldots]^{-1}]^{-1}]^{-1}]^{-1}]^{-1} \qquad (3)$$

3) The third Cauer matrix form is

$$[T(s)] = [K_1+K_1's+[K_2\frac{1}{s}+K_2'+[K_3+K_3's+[K_4\frac{1}{s}+K_4'+[\ldots]^{-1}]^{-1}]^{-1}]^{-1}]^{-1} \qquad (4)$$

where H_i', H_i, K_i and K_i' are matrix quotients of real and constant m by m matrices. The matrix quotients in Eqs. (2), (3) and (4) can be evaluated by extending the scalar Routh algorithm[6] to the matrix Routh algorithm.[7] For example, if the matrix quotients H_i in Eq. (3) are required, the following matrix Routh algorithm and matrix Routh array are applied.

$$\begin{array}{lcccccc}
H_1 = A_{11}A_{21}^{-1} & & A_{11} & A_{12} & A_{13} & A_{14} & \cdots \\
H_2 = A_{21}A_{31}^{-1} & & A_{21} & A_{22} & A_{23} & \cdots & \\
H_3 = A_{31}A_{41}^{-1} & & A_{31} & A_{32} & \cdots & & \\
 & & A_{41} & & & & \\
 & & \cdots & & & &
\end{array} \tag{5}$$

where $A_{j,\ell} = A_{j-2,\ell+1} - H_{j-2}A_{j-1,\ell+1}$ (5a)

$j = 3,4,\ldots\ ;\ \ell = 1,2,\ldots$

$$H_p = A_{p,1}(A_{p+1,1})^{-1}\ ;\ p = 1,2,\ldots \tag{5b}$$

$$\det(A_{p+1,1}) \neq 0 \tag{5c}$$

It is noted that the matrix Routh array may be terminated prematurely. When a matrix in the first column of the matrix Routh array is singular, or $\det(A_{j,1}) = 0$, the original transfer-function matrix can be multiplied by a matrix polynomial $(sI+Q)$ where Q is a constant matrix so that the condition of Eq. (5c) can be met.

REALIZATION OF TRANSFER-FUNCTION MATRIX

The realization of the transfer-function matrix of multivariable systems has been studied by numerous authors[8,9,10,11]. In this paper the dynamic equations of a transfer-function matrix are formulated by means of matrix continued fraction.

The formulation of Eq. (4) is a mixture of the first and the second Cauer matrix forms, therefore it is a general form. The block diagram corresponding to Eq. (4) is shown in Fig. 1. If the inputs to the integrators are assigned rated variables, a dynamic equation can be written as follows.

$$A_1\ddot{X}+A_2\dot{X}+A_3X = Br \tag{6}$$

$$Y = C_1^TX+C_2^T\dot{X} \tag{6a}$$

$$X(0) = [0] \text{ and } \dot{X}(0) = [0]\ , \text{ where} \tag{6b}$$

$$A_1 = \begin{bmatrix}
\boxed{(K_1')K_2'} & (K_1')K_4' & (K_1')K_6' & \cdots & (K_1')K_n' \\
(K_1')K_2' & (K_1'+K_3')K_4' & (K_1'+K_3')K_6' & \cdots & (K_1'+K_3')K_n' \\
(K_1')K_2' & (K_1'+K_3')K_4' & (K_1'+K_3'+K_5')K_6' & \cdots & (K_1'+K_3'+K_5')K_n' \\
 & & & \cdots & \\
(K_1')K_2' & (K_1'+K_3')K_4' & (K_1'+K_3'+K_5')K_6' & \cdots & (K_1'+\ldots+K_{n-1}')K_n'
\end{bmatrix}$$

$$A_2 = \begin{bmatrix} 1+(K_1')K_2+(K_1)K_2' & (K_1')K_4+(K_1)K_4' & \cdots & (K_1')K_n+(K_1)K_n' \\ (K_1')K_2+(K_1)K_2' & 1+(K_1'+K_3')K_4+(K_1+K_3)K_4' & \cdots & (K_1'+K_3')K_n+(K_1+K_3)K_n' \\ (K_1')K_2+(K_1)K_2' & (K_1'+K_3')K_4+(K_1+K_3)K_4' & \cdots & (K_1'+K_3'+K_5')K_n+(K_1+K_3+K_5)K_n' \\ (K_1')K_2+(K_1)K_2' & (K_1'+K_3')K_4+(K_1+K_3)K_4' & \cdots & 1+(K_1'+\ldots+K_{n-1}')K_n+(K_1+\ldots+K_{n-1})K_n' \end{bmatrix}$$

$$A_3 = A_1\Big|_{\text{substituting } K_i \text{ for } K_i'}$$

$$B^T = [1,1,\ldots,1]$$

$$C_1^T = [K_2,K_4,\ldots,K_n]$$

$$C_2^T = [K_2',K_4',\ldots,K_n']$$

$$X^T = [X_{11},X_{12},\ldots,X_{1,\frac{n}{2}}]$$

where $X_{11}^T = [x_1,\ldots,x_m]$

$$X_{12}^T = [x_{m+1},\ldots,x_{2m}]$$

$$X_{1,\frac{n}{2}}^T = [x_{(n-1)m+1},\ldots,x_{n\frac{m}{2}}]$$

The matrix quotients K_i and K_i' can be obtained by the following algorithm.[6]

$$A_{j,\ell} = A_{j-2,\ell+1} - K_{j-2}A_{j-1,\ell+1} - K_{j-2}'A_{j-1,\ell} \tag{6c}$$

$$j = 3,4,\ldots\ ;\ \ell = 1,2,\ldots$$

$$K_p = A_{p,1}(A_{p+1,1})^{-1}\ ;\ K_p' = A_{p,(n+2-p)}(A_{p+1,n+1-p})^{-1} \tag{6d}$$

$$\det(A_{p+1,1}) \neq 0\ ;\ \det(A_{p+1,n+1-p}) \neq 0 \tag{6e}$$

If all K_i' matrix quotients in Eqs. (4) and (6) vanish and all K_i are replaced by H_i, $i=1,2,\ldots,2j$ where $j \leq n$ in Eqs. (4) and (6) we have the dynamic equations for the second Cauer matrix form as follows.

$$\dot{X} = -A_3X+Br \tag{7}$$

$$Y = C_1^TX \tag{7a}$$

The matrix quotients H_i can be obtained from Eq. (5). The corresponding block diagram is shown in Fig. 2. If we let $\dot{X}=Z$ and $K_i=[0]$ and substitute all H_i' for K_i' in Eqs. (4) and (6) we have the state equation of the first Cauer matrix form

$$\dot{Z} = -A_1^{-1}Z + A_1^{-1}Br \tag{8}$$

$$Y = C_2^T Z \tag{8a}$$

The matrix quotients H_1' can be evaluated by the algorithm of Eq. (5) except that the matrix polynomials of Eq. (1) should be rearranged into descending order, and redefine the subscripts of the coefficient matrices into ascending order.

From Eqs. (5b) and (6d) we observe that the determinant of each matrix quotient H_i, K_i, K_i' and H_i' does not vanish or each matrix quotient is of full rank. Therefore Eqs. (6), (7) and (8) are minimal realization[8,9,10,11] of the transfer-function matrix of a multivariable system. This means Eqs. (6), (7) and (8) are completely controllable and observable dynamic equations. This conclusion can be easily verified by Gilbert's theorems.[18]

INVERSION OF RATIONAL MATRICES

The problems of inverting rational matrices whose entries are in the form of rational functions in s have been studied by Munro and Zakian[12], and Huseyin and Abdullah.[13] An algebraic method is established in this paper to obtain the inversion of rational matrices by means of matrix continued fraction. The rational matrices can be written in the form of Eq. (1) but let $A_{21} = [I]$ and $A_{2j} = [0]$. The transfer-function matrix of Eq. (1) can be expressed by a matrix continued fraction of second Cauer matrix form. The matrix quotients are obtained from Eq. (5). The corresponding state equations are then formulated in Eq. (7). Applying the Leverrier algorithm[9,14] to Eq. (7) yields the inversion of a rational matrix. The procedure can be well illustrated by Huseyin and Abdullah's example.[13]

Let $$[T(s)] = \begin{bmatrix} s^3+2s+1 & s^2 \\ s & s^3+s+1 \end{bmatrix}^{-1} = [I][A_{11}+A_{12}s+A_{13}s^2+A_{14}s^3]^{-1}$$

$$= \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\right]\left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 2 & 0 \\ 1 & 1 \end{pmatrix}s + \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}s^2 + \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}s^3\right]^{-1} \tag{9}$$

In this example if we apply the algorithm shown in Eq. (5) directly, it will result that the condition of Eq. (5c) is violated. This can be remedied by premultiplying the [T(s)] by a diagonal matrix polynomial (s+2)[I]. Eq. (9) may be modified as follows.

$$[T'(s)] = [2I+Is][T(s)]$$

$$= [A_{21}+A_{22}s][A_{11}+A_{12}s+A_{13}s^2+A_{14}s^3]^{-1} \tag{10}$$

The first two rows of the matrix Routh array are

$$\begin{array}{cccc|cccc} A_{11} & A_{12} & A_{13} & A_{14} & \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix} & \begin{pmatrix}2 & 0\\ 1 & 1\end{pmatrix} & \begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix} & \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix} \\ A_{21} & A_{22} & A_{23} & & \begin{pmatrix}2 & 0\\ 0 & 2\end{pmatrix} & \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix} & \begin{pmatrix}0 & 0\\ 0 & 0\end{pmatrix} & \end{array} \tag{11}$$

Applying the algorithm of Eq. (5) we have the matrix quotients

$$H_1 = \frac{1}{2}\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}; \qquad H_2 = \frac{1}{3}\begin{pmatrix}4 & 0\\ -8 & 12\end{pmatrix}; \qquad H_3 = \frac{1}{11}\begin{pmatrix}16.5 & 6\\ 11 & 5.5\end{pmatrix};$$

$$H_4 = \frac{1}{3}\begin{pmatrix}5.5 & -10.75\\ 0 & 5.5\end{pmatrix}; \; H_5 = \frac{4}{3531}\begin{pmatrix}-1155 & -1191\\ -528 & 627\end{pmatrix}; H_6 = \frac{1}{3}\begin{pmatrix}-9.5 & 10.75\\ 8 & -17.5\end{pmatrix} \tag{12}$$

Substituting these H_i matrices into Eq. (7) and applying the Leverrier algorithm[9,14] we have the transfer function matrix $[T'(s)]$.

$$[T'(s)] = \frac{1}{s^6+3s^4+s^3+2s^2+3s+1}\begin{bmatrix} s^4+2s^3+s^2+3s+2 & -s^3-2s^2 \\ -s^2-2s & s^4+2s^3+2s^2+5s+2 \end{bmatrix} \tag{13}$$

Renormalizing Eq. (13) by premultiplying it by the factor $1/(s+2)$ [I] we have the required $[T(s)]$.

$$[T(s)] = \frac{1}{s^6+3s^4+s^3+2s^2+3s+1}\begin{bmatrix} s^3+s+1 & -s^2 \\ -s & s^3+2s+1 \end{bmatrix} \tag{14}$$

The trivial assumption[13] of the characteristic polynomial can be avoided, and all processes are amenable to digital computation.

MODEL REDUCTION

When a high order and large dimensional multivariable system is to be analyzed, a low order reduced model is often searched for which digital or analog simulation is possible. The model reduction by continued fraction for single variable systems has been investigated by numerous authors.[15,16,17] Chen[4] applied the second Cauer matrix form to simplify a high order gas turbine system. The simplified model obtained by the second Cauer matrix form gives a satisfactory approximation in the steady-state response but not in the transient portion. The applications of the first and the third Cauer matrix forms to multivariable system reduction are new. If the first Cauer matrix form is used for model reduction, we have a good approximation in the transient portion but not in the steady-state portion. However, if the third Cauer matrix form is applied, a satisfactory approximation for both the transient and the steady-state responses can be obtained. This can be verified by the initial-value and the final-value theorems, or

$$\lim_{t\to\infty} [Y(t)] = \lim_{s\to 0} s\cdot[T(s)][R(s)] = [K^1]^{-1} = [H_1]^{-1} \tag{15}$$

when $[R(s)] = \frac{1}{s}[I]$ or unit step functions are chosen as inputs.

$$\lim_{t\to 0}[Y(t)] = \lim_{s\to\infty} s[T(s)][R(s)] = [K_1']^{-1} = [H_1']^{-1} \tag{16}$$

with impulse functions used as inputs or $[R(s)] = [I]$. From Eq. (15) and (16) we observe that the first matrix quotients K_1 and H_1 dominate the steady-state response. Whereas K_1' and H_1' contribute the initial waveform of the transient response. Therefore the first matrix quotient $K_1+K_1's$, H_1 or $H_1's$ is the most important one. The matrix quotients which are located in the inner portion of Eqs. (2), (3) and (4) or the inner loop of the block diagram in Figs. (1) and (2) are low performance terms which can be discarded without significantly effecting the performance of the system responses. Thus a low order model can be obtained to approximate the original system.

To illustrate the procedure, consider that a reduced model by the Cauer matrix form of the following transfer-function matrix is required.

$$[T(s)] = \frac{1}{\Delta(s)}\begin{bmatrix} T_{11}(s) & T_{12}(s) \\ T_{21}(s) & T_{22}(s) \end{bmatrix} \tag{17}$$

where

$\Delta(s)=s^8+7.131s^7+72.04s^6+355.8s^5+1048.9s^4+1969.4s^3+21543s^2+1068.3s+187.$

$T_{11}(s)=s^7+7.131s^6+25.34s^5+55.38s^4+72s^3+49.65s^2+14.99s+1.4$

$T_{12}(s)=s^6+6.851s^5+23.42s^4+48.82s^3+60.03s^2+36.46s+7.9$

$T_{21}(s)=-46.7s^6-300.4s^5-976.9s^4-1919.7s^3-2139.3s^2-1066.9s-187.$

$T_{22}(s)=s^7+6.851s^6+23.42s^5+48.82s^4+60.03s^3+36.46s^2+7.9s$

The reduced model is

$$[Y(s)] \doteq [K_1+K_1's+[K_2\frac{1}{s} + K_2']^{-1}]^{-1}[R(s)] \tag{18}$$

where

$$[K_1] = \begin{bmatrix} 0 & -1 \\ 23.67 & 0.1772 \end{bmatrix} \qquad [K_1'] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$[K_2] = \begin{bmatrix} -193142.1 & 0.03875 \\ 62329847 & -0.00006086 \end{bmatrix} \qquad [K_2'] = \begin{bmatrix} -2.712\times 10^{11} & 0.1076 \\ 6.074\times 10^{14} & -14.38 \end{bmatrix} \tag{18a}$$

The corresponding state equation of the reduced model can be obtained by partitioning the system matrices in Eq. (6), or

$$[K_1'K_2'][\ddot{X}_{11}]+[I+K_1'K_2+K_1K_2'][\dot{X}_{11}]+[K_1K_2][X_{11}] = [I]r$$
$$[Y] = [K_2][X_{11}]+[K_2'][\dot{X}_{11}] \tag{19}$$

The transfer-function matrix of reduced model $[T'(s)]$ can be obtained from Eq. (18) directly or indirectly taking the Laplace transform of Eq. (19).

$$[T'(s)] = [K_2+K_2's][K_1K_2+(I+K_1'K_2+K_1K_2')s+K_1'K_2's^2]^{-1}$$

$$= \frac{1}{\Delta'(s)}\begin{bmatrix} T_{11}'(s) & T_{12}'(s) \\ T_{21}'(s) & T_{22}'(s) \end{bmatrix} \tag{20}$$

where

$$\Delta'(s) = s^4+1.172s^3+46.86s^2+21.12s+0.2167\times10^{-4}$$

$$T_{11}'(s) = s^3+1.172s^2+0.1581s+0.1622\times10^{-6}$$

$$T_{12}'(s) = s^2+0.892s+0.9157\times10^{-6}$$

$$T_{21}'(s) = -46.7s^2-21.12s-0.2167\times10^{-4}$$

$$T_{22}'(s) = s^3+0.8923s^2+0.9157\times10^{-6}s$$

If impulse functions are used as inputs, the output responses of the original system and the reduced model are written as follows.

1) The original system

$$Y_1(s) = \frac{1}{\Delta(s)}[s^7+8.131s^6+32.19s^5+78.8s^4+120.82s^3+109.68s^2+51.45s+9.3] \tag{21}$$

$$Y_2(s) = \frac{1}{\Delta(s)}[s^7-39.85s^6-277s^5-928.12s^4-1859.72s^3-2102.83s^2-1059.03s-186.99] \tag{21a}$$

The $\Delta(s)$ is shown in Eq. (17).

2) The reduced model

$$Y_1'(s) = \frac{1}{\Delta'(s)}[s^3+2.172s^2+1.05s+1.078\times10^{-6}] \tag{22}$$

$$Y_2'(s) = \frac{1}{\Delta'(s)}[s^3-45.81s^2-21.12s-2.167\times10^{-5}] \tag{22a}$$

The $\Delta'(s)$ is shown in Eq. (20).

The time response curves of the original system are compared with the approximated model in Fig. 3. The approximation is very satisfactory. It should be pointed out that a stable system may yield unstable reduced model because the continued fraction method of approximation does not necessarily give a stable model. The statiblity of reduced model can be determined by checking the locations of the eigenvalues of the characteristic equation obtained from partitioned dynamic equations in Eqs. (6), (7), and(8). A low order stable model is usually chosen to approximate the original high order system.

CONCLUSION

The formulation of continued fraction for single variable systems has been extended to the matrix continued fraction for multivariable systems. The realization of transfer-function matrices has been obtained by the matrix continued fractions. The applications of the Cauer matrix forms have been extended to linear model reduction for multivariable systems. An algebraic method has been established to invert rational metrices whose entries are rational polynomial in s. Some properties of the matrix continued fraction of three Cauer matrix forms have been presented.

ACKNOWLEDGMENT

The authors wish to express their gratitude for the helpful discussion of Professor C. F. Chen of the University of Houston. This work was supported in part by the National Science Foundation Grant GK-42110, and in part by the Army Research Office Grant DAHCO 4-74-G-0145.

REFERENCES

[1] L. Weinberg, Network Analysis and Synthesis, New York: McGraw-Hill, pp. 193-199, 1962.

[2] W. D. Fryer, "Applications of Routh's Algorithm to Network Theory Problem," IRE Trans. Circuit Theory, Vol. CT-6, pp. 144-149, June '59.

[3] C. F. Chen and L. S. Shieh, "An Algebraic Method for Control System Design," Internal J. Contr., Vol. 11, pp. 717-739, 1970.

[4] C. F. Chen, "Model Reduction of Multivariable Control Systems by Means of Matrix Continued Fraction," in Preprints 5th IFAC, pp.35.1-35.8,'72.

[5] L. S. Shieh, J. M. Navarro, and R. Yates, "Multivariable Systems Identification Via Frequency Responses," proceedings IEEE (letter), Vol. 62, No. 8, 1974.

[6] L. S. Shieh and M. J. Goldman, "Continued Fraction Expansion and Inversion of the Cauer Third Form" IEEE Trans. Circuits and Systems, Vol. CAS-21, pp.341-345, May 1974.

[7] L. S. Shieh and F. F. Gaudiano, "Matrix Continued Fraction Expansion and Inversion by the Generalized Matrix Routh Algorithm," To appear in Internal. J. Contr., 1975.

[8] L. M. Silverman, "Realization of Linear Dynamic Systems," IEEE Trans., Vol. AC-16, No. 6, pp. 554-567, Dec. 1971.

[9] C. T. Chen, Introduction to Linear System Theory, New York: Holt, Rinehart and Winston, pp. 216-254, 1970.

[10] R. T. N. Chen, "Transfer-Characterization and the Unique Realization of Linear Time-Invariant Multivariable Systems," JACC Preprint 1972.

[11] B. L. Ho and R. E. Kalman, "Effective Construction of Linear State-Variable Models from Input Output Data," Proc. 3rd Annual Allerton Conference, pp. 449-459, 1966.

[12] N. Munro and V. Zakian, "Inversion of Rational Polynomial Matrices," Electron. Lett., Vol. 6, No. 19, 1970.

[13] E. E. O. Huseyin and K. Abdullah, "On the Inversion of Rational Matrices," IEEE Trans. Circuits and Systems, Vol. CAS-21, No. 1, Jan. pp. 8-10, 1974.

[14] L. A. Zadeh and C. A. Desoer, Linear System Theory, New York: McGraw-Hill, 1963.

[15] C. F. Chen and L. S. Shieh, "A Novel Approach to Linear Model Simplifications," Int. J. Contr., Vol. 8, No. 6, pp. 561-570, 1968.

[16] J. Schwartz and T. F. Edgar, "The Development of Simplified Adaptive Control Models Using the Continued Fraction Expansion," JACC Preprint, pp. 484-496, 1974.

[17] L. S. Shieh and M. J. Goldman, "A Mixed Cauer Form for Linear System Reduction," To appear in IEEE Trans. Systems, Man and Cybernetics, Nov. 1974.

[18] E. G. Gilbert, "Controllability and Observability in Multivariable Control Systems," SIAM J. Control, Vol. 1, pp. 128-151, 1963.

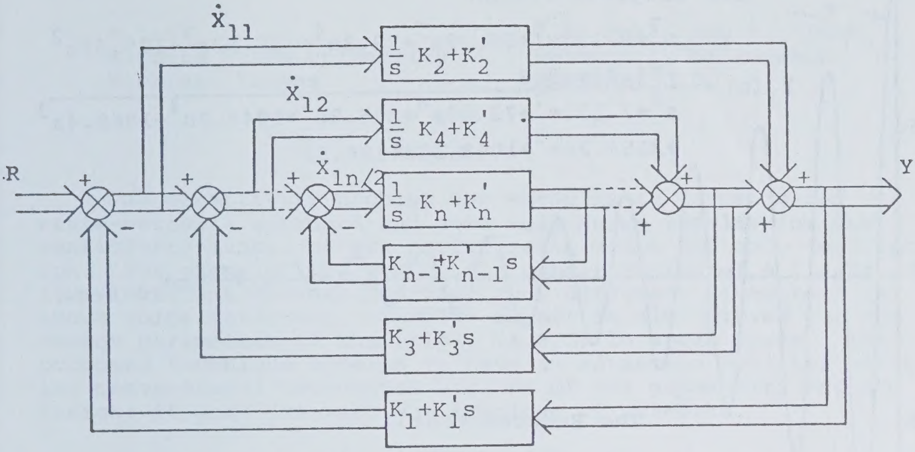

Ẋ11
1/s K2+K2'
Ẋ12
1/s K4+K4'
R
Ẋ1n/2
1/s Kn+Kn'
Y
Kn-1+K'n-1 s
K3+K3's
K1+K1's
+
-

Figure 1. Block Diagram Representation of Third Cauer Matrix Form.

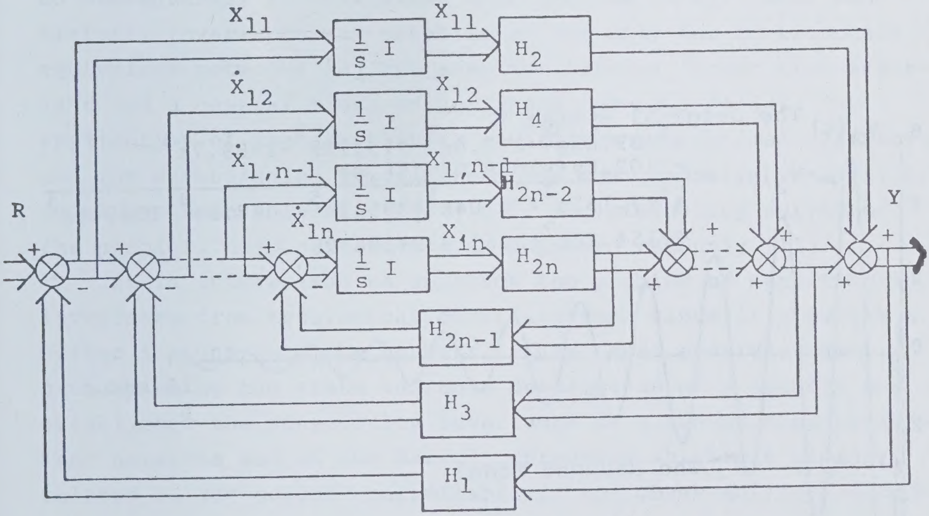
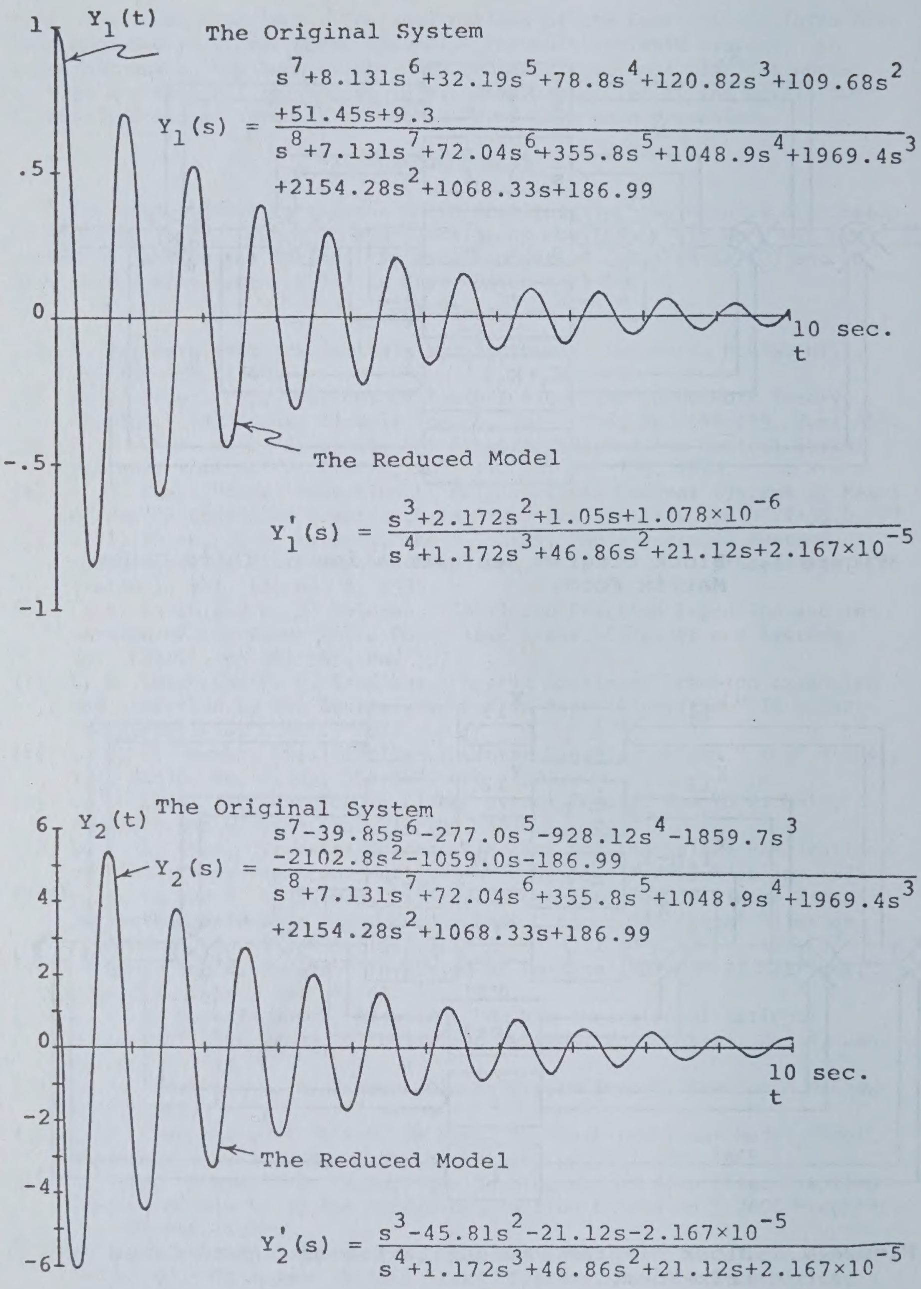

Ẋ11
1/s I
X11
H2
Ẋ12
X12
H4
Ẋ1,n-1
X1,n-1
H2n-2
R
Ẋ1n
X1n
H2n
Y
H2n-1
H3
H1
+
-

Figure 2. Block Diagram Representation of Second Cauer Matrix Form.



$$Y_1(s) = \frac{s^7+8.131s^6+32.19s^5+78.8s^4+120.82s^3+109.68s^2+51.45s+9.3}{s^8+7.131s^7+72.04s^6+355.8s^5+1048.9s^4+1969.4s^3+2154.28s^2+1068.33s+186.99}$$

$$Y_1'(s) = \frac{s^3+2.172s^2+1.05s+1.078\times10^{-6}}{s^4+1.172s^3+46.86s^2+21.12s+2.167\times10^{-5}}$$

$$Y_2(s) = \frac{s^7-39.85s^6-277.0s^5-928.12s^4-1859.7s^3-2102.8s^2-1059.0s-186.99}{s^8+7.131s^7+72.04s^6+355.8s^5+1048.9s^4+1969.4s^3+2154.28s^2+1068.33s+186.99}$$

$$Y_2'(s) = \frac{s^3-45.81s^2-21.12s-2.167\times10^{-5}}{s^4+1.172s^3+46.86s^2+21.12s+2.167\times10^{-5}}$$

Figure 3. Comparison of Time Responses

SENSITIVITY INVARIANCE USING STATE VARIABLES

M. LAL
Wichita State University
Wichita, Kansas

J.S. SOHAL AND H. SINGH
University of Roorkee
Roorkee, U.P.

ABSTRACT

The sensitivity problem for networks is investigated via state-variable approach and some well known results for the sensitivity functions are proved using state variable description. For example, the sum of the sensitivities of a linear time-invariant network function over different parameters is shown to be invariant. Similar result is also proved for the Markov parameters of the triple (A,B,C) in state space. The proposed technique appears to have an advantage over the existing conventional techniques because of its generality and the insight it provides into the topology of networks.

I. INTRODUCTION

Leeds and Ugran [1] proposed that the sum of the sensitivity functions over all components of each continuously equivalent networks, defined by Schoeffler [2], is invariant with respect to various such networks. A large number of papers appeared subsequently investigating this problem [3-8]. This sensitivity invariance was established not only for continuously equivalent networks [3] but also for general linear time-invariant and a general class of nonlinear networks [4-8]. The application of the sensitivity invariance for optimal synthesis and for establishing lower bounds for the quadratic sensitivity index has been shown in [9], [10]. An interesting survey on the problems of sensitivity also appeared recently [11].

It is interesting to approach the problem of sensitivity invariance from topological considerations since it provides a better insight into the network. This paper considers the problem using the state variable description of a network and establishes the sensitivity invariance of a linear time-invariant networks and of the Markov paprmeters which are closely related to the output controllability and observability, etc.

II. STATE-VARIABLE DESCRIPTION AND SENSITIVITY INVARIANTS

The state model of a general linear time-invariant network is given by

$$\dot{X} = AX + BU$$
$$Y = CX + DU \qquad (1)$$

where X is the nx1 state vector, u is v x 1 input vector, y is mx1 output vector and A,B,C,D are state-model matrices having constant elements for LTIV networks, obtained from the topology and the elements of the network [12]. The matrices A and B contain the dynamic elements of the network and can be written as [13].

$$A = \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array}\right] = \left[\begin{array}{c|c} \mathcal{C}^{-1} & 0 \\ \hline 0 & \mathcal{L}^{-1} \end{array}\right] \left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array}\right]$$

$$B = \left[\begin{array}{c|c} B_{11} & B_{12} \\ \hline B_{21} & B_{22} \end{array}\right] = \left[\begin{array}{c|c} \mathcal{C}^{-1} & 0 \\ \hline 0 & \mathcal{L}^{-1} \end{array}\right] \left[\begin{array}{c|c} N_{11} & N_{12} \\ \hline N_{21} & N_{22} \end{array}\right]$$

where $\mathcal{C} = C_C + F'_{SC} C_S F_{SC} = D_C^{-1} + F'_{SC} D_S^{-1} F_{SC}$

$$\mathcal{L} = L_L + F_{L\Gamma} L_\Gamma F'_{L\Gamma} \tag{2}$$

and M_{ij}, N_{ij}, C, D contain information about resistive part of the network and its topology.

Let us consider single-input (voltage source) single-output (current) LTIV network for the sake of simplicity and also assume that the voltage source does not form a loop with capacitors only, although pure capacitor loops and inductor outsets are permissible. The state-medel for such a network can be easily derived using the notation in [12] and approach in [13].

$$\begin{aligned} \dot{X} &= AX + bu \\ y &= cX + du \end{aligned} \tag{3}$$

where $X = \begin{bmatrix} v_C \\ i_L \end{bmatrix}$, y and u are the output current and input voltage respectively, and

$$M_{11} = -F'_{RC}\,\mathcal{R}^{-1} F_{RC}$$

$$M_{12} = F'_{LC} - F'_{RC}\,\mathcal{R}^{-1} F_{RS} R_G F'_{LG}$$

$$M_{21} = -F_{LC} + F_{LG}\,\mathcal{G}^{-1} F'_{RG} R_R^{-1} F_{RC}$$

$$M_{22} = -F_{LG}\,\mathcal{G}^{-1} F'_{LG}$$

$$\mathcal{R} = R_R + F_{RG} R_G F'_{RG}$$

$$\mathcal{G} = G_G + F'_{RG} G_R F_{RG} = R_G^{-1} + F'_{RG} R_R^{-1} F_{RG}$$

$$[b] = \begin{bmatrix} b_1 \\ \hdashline b_2 \end{bmatrix} = \begin{bmatrix} \mathcal{C}^{-1} & \\ & \mathcal{L}^{-1} \end{bmatrix} \begin{bmatrix} -F'_{RC}\,\mathcal{R}^{-1}\,F_{RV} \\ \hdashline -F_{LV} + F_{LG}\,\mathcal{G}^{-1}\,F_{RG}\,R_R^{-1}\,F_{RV} \end{bmatrix} = \begin{bmatrix} \mathcal{C}^{-1} & \\ & \mathcal{L}^{-1} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

$$[c] = \left[c_1 \;\vdots\; c_2 \right] = \left[F'_{RV}\,\mathcal{R}^{-1} F_{RV} \;\vdots\; -F'_{LV} + F'_{RV}\,\mathcal{R}^{-1}\,F_{RG}\,R_G\,F'_{LG} \right]$$

$$d = F'_{RV}\,\mathcal{R}^{-1}\,F_{RV} \tag{4}$$

The subscript C, G, Γ correspond to the tree branches, S,R,L correspond to the links, ´ denotes the transpose, and F_{RG} etc., are the fundamental circuit submatrices giving the topological relations between the various branches of the graph.

The following matrix relationship will be used extensively in developing the sensitivity analysis [14].

$$\frac{\partial P^{-1}}{\partial t} = -P^{-1}\,\frac{\partial P}{\partial t}\,P^{-1} \tag{5}$$

For example, while considering the variation in $\mathcal{R}^{-1}$ due to changes in the diagonal matrix elements of R_R and R_S, we have using (5)

$$\frac{\partial \mathcal{R}^{-1}}{\partial R_{R_j}} = -\mathcal{R}^{-1}\,\frac{\partial \mathcal{R}}{\partial R_{R_j}}\,\mathcal{R}^{-1}$$

and

$$\sum_{R_{R_j}} R_{R_j}\,\frac{\partial \mathcal{R}^{-1}}{\partial R_{R_j}} = \mathcal{R}^{-1}\,R_R\,\mathcal{R}^{-1}$$

Similarly,

$$\sum_{R_{G_j}} R_{G_j}\,\frac{\partial \mathcal{R}^{-1}}{\partial R_{G_j}} = -\mathcal{R}^{-1}\,(F_{RG}\,R_G\,F'_{RG})\,\mathcal{R}^{-1}$$

and thus

$$\sum_{q_j=\{R_{R_j},R_{G_j}\}} q_j\,\frac{\partial \mathcal{R}^{-1}}{\partial q_j} = -\mathcal{R}^{-1}\,\mathcal{R}\,\mathcal{R}^{-1} = -\mathcal{R}^{-1} \tag{6}$$

We can easily show in the same manner that for $q_j = \{R_{R_j}, R_{G_j}\}$

$$\sum_{q_j} q_j\,\frac{\partial \mathcal{G}^{-1}}{\partial q_j} = \mathcal{G}^{-1}$$

and for $p_i = \{D_{C_i}, D_{S_i}, L_{L_i}, L_{T_i}\}$

$$\sum_{p_i} p_i \frac{\partial \mathcal{C}^{-1}}{\partial p_i} = \mathcal{C}^{-1} \tag{7}$$

$$\sum_{p_i} p_i \frac{\partial \mathcal{L}^{-1}}{\partial p_i} = -\mathcal{L}^{-1} \tag{8}$$

It can be easily shown now that the sum of the sensitivities for the network function W(s) (admittance function in this case) over all the components of a given network is invariant. The network function in terms of (A,b,c,d) can be written as

$$W(s) = c(sI-I)^{-1} b + d \tag{9}$$

and let

$$\sum_{e_k} e_k \frac{\partial W}{\partial e_k} = \sum_{p_i} p_i \frac{\partial W}{\partial p_i} + \sum_{q_j} q_j \frac{\partial W}{\partial q_j} \tag{10}$$

where $p_i = \{D_{c_i}, D_{s_i}, L_{L_i}, L_{\Gamma_i}\}$, $q_j = \{R_{R_j}, R_{G_j}\}$.

From (9),

$$\sum_{p_i} p_i \frac{\partial W}{\partial p_i} = c \sum p_i \frac{\partial}{\partial p_i} (sI-A)^{-1}.b + c\,(sI-A)^{-1}\{\Sigma p_i \frac{\partial b}{\partial p_i}$$

Using (4), (5), (7), and (8) we can write

$$\sum_{p_i} p_i \frac{\partial W}{\partial p_i} = -c(sI-A)^{-1} \{\Sigma p_i \frac{\partial}{\partial p_i} (sI-A)\}\ (sI-A)^{-1} b$$

$$+ c(sI-A)^{-1} \sum p_i \frac{\partial b}{\partial p_i}$$

$$= c(sI-A)^{-1} \begin{bmatrix} A_{11} & A_{12} \\ -A_{21} & -A_{22} \end{bmatrix} + (sI-A)^{-1} b + c\ sI-A)^{-1} \begin{bmatrix} b_1 \\ -b_2 \end{bmatrix} \tag{11}$$

From (4), (5), (6), (9),

$$\sum_{qj} q_i \frac{\partial W}{\partial q_j} = c \sum q_i \frac{\partial (sI-A)^{-1}}{\partial q_j} \cdot b + \Sigma q_i \frac{\partial c}{\partial q_j} (sI-A)^{-1} b$$

$$+ c(sI-A)^{-1} \Sigma q_j \frac{\partial b}{\partial q_j} + \Sigma q_j \frac{\partial d}{\partial q_j}$$

$$= c(sI-A)^{-1} \begin{bmatrix} -A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} (sI-A)^{-1}b + [-c_1 \ \ 0](sI-A)^{-1}b$$

$$+ c(sI-A)^{-1} \begin{bmatrix} -b_1 \\ 0 \end{bmatrix} - d \qquad (12)$$

Substituting from (11) and (12) in (10) we have,

$$\sum_{e_k} e_k \frac{\partial W}{\partial e_k} = c(sI-A)\begin{bmatrix} 0 & A_{12} \\ -A_{21} & 0 \end{bmatrix} (sI-A)^{-1}b$$

$$- [c_1 \ \ 0](sI-A)^{-1}b - c(sI-A)^{-1}\begin{bmatrix} b_1 \\ 0 \end{bmatrix} -d \qquad (13)$$

Let

$(sI-A)^{-1} = \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix}$. Substituting in the second and third terms on the right hand side of (12) and expanding, we get

$$\sum_{e_k} e_k \frac{\partial W}{\partial e_k} = c(sI-A)^{-1} \begin{bmatrix} 0 & A_{12} \\ -A_{21} & 0 \end{bmatrix} (sI-A)^{-1} b - c \begin{bmatrix} F_1 & 2F_2 \\ 0 & F_4 \end{bmatrix} b - d$$

$$= \{c(sI-A)^{-1} \begin{bmatrix} 0 & A_{12} \\ -A_{21} & 0 \end{bmatrix} (sI-A)^{-1} b - c \begin{bmatrix} 0 & F_2 \\ -F_3 & 0 \end{bmatrix} b\}$$

$$- c \ (sI-A)^{-1} b - d$$

$$= c(sI-A)^{-1} \{ \begin{bmatrix} 0 & A_{12} \\ -A_{21} & 0 \end{bmatrix} \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix} - (sI-A) \begin{bmatrix} 0 & F_2 \\ -F_3 & 0 \end{bmatrix} \} .b$$

$$- W(s) \qquad (14)$$

On expanding the terms inside { } in (14) and remembering that [F] in the inverse of (sI-A), we notice that these terms add to zero and hence,

$$\sum_{e_k} e_k \frac{\partial W}{\partial e_k} = - W(s)$$

or

$$S^{W}_{e_k} \triangleq \sum_{e_k} \frac{e_k}{W} \frac{\partial W}{\partial e_k} = -1 \tag{15}$$

Similar results for any other network function can be easily proved. The generalization to multi-part networks, and higher order sensitivity invariance results can also be obtained.

The sensitivity invariance of Markov parameters defined as $c\,A^{n}b$ $(n = 0,1,2,\ldots)$, can also be established as follows. $W(s)$ can be written as

$$W(s) = \sum_{h=0}^{\infty} cA^{n}b\, s^{-n-1} + d$$

$$= \sum_{n=0}^{\infty} W_n s^{-n-1} + d \tag{16}$$

For $W_0 = cb$,

$$\sum_{p_i} p_i \frac{\partial W_0}{\partial p_i} = c \sum_{p_i} p_i \frac{\partial b}{\partial p_i}$$

$$= c \begin{bmatrix} b_1 \\ -b_2 \end{bmatrix} = c_1 b_1 - c_2 b_2$$

$$\sum_{q_j} q_j \frac{\partial W_0}{\partial q_j} = \sum q_j \frac{\partial c}{\partial q_j} \cdot b + c \sum q_j \frac{\partial b}{\partial q_j}$$

$$= [-c_1 \;\; 0]\, b + c \begin{bmatrix} -b_1 \\ 0 \end{bmatrix} = -2\, c_1 b_1$$

$$S^{W_0}_{e_k} = \frac{1}{W0} (-c_1 b_1 - c_2 b_2) = -1$$

Similiarly for W_1 = CAb

$$\sum_{e_k} e_k \frac{\partial W1}{\partial e_k} = c \begin{bmatrix} A_{11} & A_{12} \\ -A_{21} & -A_{22} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} + cA \begin{bmatrix} b_1 \\ -b_2 \end{bmatrix} + [-c_1 \;\; 0]\, Ab + cA \begin{bmatrix} -b_1 \\ 0 \end{bmatrix}$$

$$+ c \begin{bmatrix} -A_1 & 0 \\ 0 & A_{11} \end{bmatrix} b$$

$$= -cAb = -W_1 \text{ etc.}$$

Proceeding in a similiar manner we can show the result in (5).

$$\sum_{e_k} e_k \frac{\partial W}{\partial e_k} = \sum_{n=0}^{\infty} \sum_{e_k} e_k \frac{\partial W_n}{\partial e_k} s^{-n-1} + \sum_{e_k} e_k \frac{\partial d}{\partial e_k}$$

$$= - \sum_{n=0}^{\infty} W_n s^{-n-1} - d$$

$$= - W(s).$$

III. CONCULSIONS

The state variable approach has been discussed for the sensitivity analysis. This approach not only leads to the known results but also provides a better insight and perhaps, a basis for the sensitivity analysis of time-varying and non-linear networks using the well developed tools of modern system theory. Some encouraging results have already been obtained by the author for continously equivalent networks, adjoint networks, and for the other measures of sensitivity for optimal design.

ACKNOWLEDGMENT

The author wishes to thank Professor William R. Ford at Wichita State University, Wichita, Kansas for providing the research facilities and encouragement.

REFERENCES

[1]. J. V. Leeds, Jr., and G. I. Ugron; "Simplified Multiple Parameter Sensitivity Calculations and Continuously Equivalent Networks," IEEE Trans. Circuit Theory, Vol. CT-14, pp. 188-191, June, 1967.

[2]. J.D. Schoeffler, "The Synthesis of Minimum Sensitivity Networks," IEEE Trans. Circuit Theory, Vol. CT-11, pp. 271-276, June, 1964.

[3] E.S. Kuh and L.G. Lau, "Sensitivity Invariants of Continuously Equivalent Networks," IEEE Trans. Circuit Theory, Vol. CT-15, pp. 175-177, Sept., 1968.

[4] M.L. Blostein, "Sensitivity Analysis of Parasitic Effects in Resistance-Terminated LC Two-parts," IEEE Trans. Circuit Theory, Vol. CT-19, pp. 21-25, Mar., 1967.

[5]. M. Sablatash and R. Seviora, "Sensitivity Invariance for Scattering Matrices," IEEE Trans. Circuit Theory (Corresp) Vol. CT-18, pp. 282-284, March, 1971.

[6]. S.W. Director and R.A. Rohrer, "The Generalized Adjoint Network and Network Sensitivities, "IEEE Trans. Circuit Theory, Vol. CT-16, pp. 318-323, Aug., 1969.

[7]. M.N.S. Swamy, C. Bhushan, and K. Thulsiraman, "Sensitivity Invariance for Nonlinear Networks, "IEEE Trans. Circuit Theory, Vol. CT-19, pp. 599-606, Nov., 1972.

[8]. M.N.S. Swamy, C. Bhushan, and K. Thulsiraman, "Sensitivity Invariance for Linear Time-Invariant Networks," IEEE Trans. Circuit Theory, Vol. CT-20, pp. 21-24, Jan., 1973.

[9]. M.N.S. Swamy, C. Bhushan, and K. Thulsiraman, "Bounds on the Sum of Sensitivity Magnitudes for Network Functions," Proc., 1972, Int. Symp. Circuit Theory , pp. 21-25

[10]. C. Schmidt and R. Kasper, "On Minimum Sensitivity Networks," IEEE Trans. Circuit Theory, Vol. CT-14, pp. 438-440, Dec., 1967.

[11]. S.R. Parker, "Sensitivity: Old Questions, Some New Answers," IEEE Trans. Circuit Theory, Vol. CT-18, pp. 27-35, Jan., 1971.

[12]. E.S. Kuh and R.A. Rohrer, "The State Variable Approach to Network Analysis," Proc. IEEE, Vol. 53, pp. 672-686, July, 1965.

[13]. R.A. Rohrer, "Circuit Theory: An Introduction to State-Variable Approach," McGraw-Hill, 1970.

[14]. K. Cgata, "State Space Analysis of Control Systems," pp. 65, Prentice Hall, 1967.

UNIDIRECTIONAL RLC TRANSMISSION LINES*

V. BURKE†, R.J. DUFFIN††, D. HAZONY†
† Electrical Engineering and Applied Physics
Case Western Reserve University
Cleveland, Ohio, 44106
††Department of Mathematics
Carnegie-Mellon University
Pittsburgh, Pennsylvania, 15213

ABSTRACT

Of concern are mechanical or electrical waves in media which may be nonuniform and dissipative. The problem posed is to find conditions for the undistorted propagation of signals. The electrical transmission line is chosen as the general model. Along the length of the transmission line there are four functions which may be prescribed essentially arbitrarily. These are series resistance, series inductance, shunt conductance, and shunt capacitance. A differential equation is derived, relating these functions, which gives a requisite for distortionless transmission of a voltage wave at least in one direction. Various corollaries of this theorem are developed. For instance conditions are derived for distortionless voltage wave propagation in both directions, and that simultaneous voltage and current waves may propagate along the line if and only if the characteristic impedance is positive at each point.

1. Introduction. Distortionless signalling has been of fundamental importance in applied science. In this note we give a fairly general characterization of when this is possible. The simplest case of distortionless signalling arises in the equation of the vibrating string.

$$V_{xx} = \omega^2 V_{tt} \quad , \quad V_x = \partial V/\partial x,\ V_t = \partial V/\partial t\,. \tag{1.1}$$

This equation has the general solution

$$V(t,x) = \phi(t-\omega x) + \psi(t+\omega x), \tag{1.2}$$

where ϕ and ψ are arbitrary functions. Thus $\phi(t-\omega x)$ is a wave traveling to the right with velocity $1/\omega$. Similarly $\psi(t+\omega x)$ is a wave traveling to the left. This solution is of great practical importance because this wave is transmitted to a distant point x without distortion. There is, of course, a delay time $\tau=\omega x$.

The equation for spherical waves in n dimensions is

$$V_{rr} + \frac{n-1}{r} V_r = \omega^2 V_{tt} \quad . \tag{1.3}$$

If n = 3 the general solution of this equation is

$$V(t,r) = \frac{\phi(t- r)}{r} + \frac{\psi(t+\omega r)}{r} \quad . \tag{1.4}$$

Again it is seen that a wave can be transmitted without distortion in both directions. But there is an attenuation inversely proportional to the distance r. It is a very happy circumstance that we live in three-dimensional space because signals can be broadcast without distortion only if

---

*Work partially supported by Research Grant DA-ARO-D-31-124-73 G131, Army Research Office, Durham, North Carolina, and by National Science Foundation Grant GK 42720.

$n = 1$ or $n = 3$. This has been discussed by R. Courant [1].

Waves can be of mechanical or electrical nature. In this paper we take as a general model the electrical transmission line. The line can be inhomogeneous (nonuniform) and dissipative. Then the series resistance r, series inductance ℓ, shunt capacitance c, and shunt conductance g per unit length may be functions of the position x on the transmission line. The model is idealized by regarding the distributed parameters to be a function only of x and not a function of the frequency. Thus physical phenomena as the skin effect and dielectric dispersion are not represented.

Applying Kirchhoff's laws gives the basic transmission line equations:

$$- V_x = r(x)\, I + \ell(x)\, I_t, \tag{1.5a}$$

$$- I_x = g(x)\, V + c(x)\, V_t, \tag{1.5b}$$

Here $V(t,x)$ is the voltage and $I(t,x)$ is the current. We proceed to operate on both sides of (1.5a) with $(r + \ell\partial/\partial t)\partial/\partial x$. The current is then eliminated by use of (1.5b). This shows that the voltage satisfies:

$$[(r+\ell\partial/\partial t)\partial^2/\partial x^2 - (r'+\ell'\partial/\partial t)\partial/\partial x - (r+\ell\partial/\partial t)^2(g+c\partial/\partial t)]\ V = 0\ . \tag{1.6}$$

This may be termed the general equation of telegraphy. The wave equations (1.1) and (1.3) are seen to be special cases of the equation of telegraphy. A prime denotes d/dx.

Considering wave propagation in one direction, we shall say that the transmission line is distortionless if given an arbitrary function $\phi(t)$ there is a solution of the form

$$V(t,x) = f(x)\ \phi[t-\tau(x)], \tag{1.7a}$$

with

$$V(t,0) = \phi(t). \tag{1.7b}$$

Here the function f is termed the attenuation and the function τ is termed the delay time. Of course f and τ are to be independent of ϕ. Also, $f(0) = 1$ and $\tau(0) = 0$.

Heaviside [2] originated the concept of distortionless transmission in a dissipative line. He proved that the condition for no distortion is $rc - g\ell = 0$, provided r, c, g and ℓ are constant. Thus Heaviside solved the problem for a homogeneous line. An exponential line has the parameters r, c, g, and ℓ tapering exponentially. Recently Ghosh [3] has shown that the exponential line is distortionless if the tapering is suitably chosen.

Our goal is to characterize the general distortionless line. The above examples will then be special cases of our characterization. In what follows it is assumed that $\ell > 0$ and $c > 0$ for basic physical reasons. However, we do allow the idealizations $r = 0$ and $g = 0$. Furthermore, any attenuation function is assumed to be non-zero. Thus we are permitted to divide by ℓ, c, and $\zeta = \sqrt{\ell/c}$, and any attenuation function. The functions r, ℓ, c, and g are assumed continuously differentiable throughout the paper. Proofs will be given in a later publication.

Regarding the distributed parameters r, ℓ, c, and g as functions of x permits considerable variation in the properties of some of the derived functions of a transmission line. For example:

(a) The delay time may be a nonlinear function of x.

(b) It is possible to have distortionless transmission in one direction but not the other. In this case the transmission line is termed unidirectional.

(c) The characteristic impedance is defined by the expression V=ZI with the understanding that waves propagate in one direction only. This impedance is characteristic of the distributed parameters of the line and, consequently, may be a function of x. Thus we refer to a characteristic impedance function.

These points are considered in subsequent sections.

A great deal of study has been given to inhomogeneous lines by many authors. Some selected references pertaining to impedance transformation, filtering, signal dispersion, and acoustics are given. [4], [5], [6], [7], [8].

2. Inductance, capacitance, and impedance of a line. There are standard procedures from potential theory for calculation of c, the capacitance per unit length of a transmission line. Also there are standard procedures from potential theory for calculating ℓ, the inductance per unit length. For parallel wires in a vacuum this well known calculation gives that $1/\sqrt{\ell c}$ is equal to the velocity of light. Likewise for a coaxial cable, $1/\sqrt{\ell c}$ is the velocity of light. The question then arises whether or not ℓc is a constant independent of cross section of the transmission line. In previous papers [9], [10], it was shown by use of Rayleighs Reciprocal Theorem, that in a vacuum ℓc is an invariant. No use of Maxwell's theory was made to establish this fact.

By inserting dielectric material between the conductors of the transmission line it is possible to increase c. By inserting magnetically permeable material it is possible to increase ℓ. In such ways it is possible to have ℓc be an arbitrary function along the line subject to the restriction

$$0 < 1/\sqrt{\ell c} \le \text{velocity of light.}$$

The upper bound here would follow from the special theory of relativity.

Pierce [11] has shown that the impedance of lossless transmission lines satisfies a Riccati equation. An extension of this result to include losses is given.

Lemma 2.1 A general transmission line has a right-characteristic impedance function which satisfies the Riccati differential equation

$$dZ/dx = [g(x) + sc(x)]Z^2 - [r(x)+s\ell(x)], \tag{2.1}$$

except possibly at isolated points.

Here $s = 2\pi i\nu$ and ν is the frequency. An interesting special case occurs when the voltage and current are proportional at each point x on the transmission line. Then the impedance is a function of x only and not of the frequency ν. When $Z = Z(x) > 0$ the impedance function is termed positive.

Theorem 2.1 A transmission line has a positive right-characteristic impedance function Z(x) if and only if

$$d\zeta/dx = g(x)\zeta^2 - r(x), \tag{2.2}$$

with

$$\zeta(x) = \sqrt{\ell/c} \quad . \tag{2.3}$$

Moreover, Z is uniquely defined as

$$Z = \zeta \quad . \tag{2.4}$$

Corresponding results for left-characteristic impedance functions are obtained from (2.1) and (2.2) by the transformation $dx \to -dx$.

3. *The general distortionless line*. We now require that the voltage V be distortion free. We do not require that I be distortion free. Again let

$$V(t,x) = f(x)\ \phi\ [t-\tau(x)]. \tag{3.1}$$

The following voltage derivatives are obtained from (3.1).

$$V_x = f'\phi - f\ \tau'\phi_t \tag{3.2a}$$

$$V_{xx} = f''\phi - (2f'\tau' + f\tau'')\phi_t + f(\tau')^2\phi_{tt} \tag{3.2b}$$

Substituting into the equation of telegraphy (1.6) gives

$$\begin{aligned}[\ell(\tau')^2 - \ell^2 c]\phi_{ttt} &+ [-2\ell\tau' f' - (2rc\ell - r(\tau')^2 + \ell^2 g + \ell\tau'' - \ell'\tau')f]\phi_{tt} \\ &+ [\ell f'' - (2r\tau' + \ell')f' - (r\tau'' - r'\tau' + 2rg\ell + r^2 c)f]\phi_t \\ &+ [rf'' - r'f' - r^2 gf]\phi = 0.\end{aligned} \tag{3.3}$$

Since ϕ is an arbitrary signal the coefficients of ϕ_{ttt}, ϕ_{tt}, ϕ_t, and ϕ must all vanish. This proves the following:

Lemma 3.1. *The general transmission line is voltage distortionless if and only if there is a function $f(x)$, not identically zero, satisfying the differential equations*:

$$-2\ell\tau' f' = [2rc\ell - r(\tau')^2 + \ell^2 g + \ell\tau'' - \ell'\tau']f, \tag{3.4}$$

$$\ell f'' - (2r\tau' + \ell')f' = (r\tau'' - r'\tau' + 2rg\ell + r^2 c)f, \tag{3.5}$$

$$rf'' = r'f' + r^2 g\ f. \tag{3.6}$$

Moreover, the delay time for waves propagating to the right is given by

$$\tau' = \sqrt{\ell c}\ , \tag{3.7}$$

while the delay time for waves propagating to the left is given by

$$\tau' = -\sqrt{\ell c}\ . \tag{3.8}$$

For waves propagating to the right, the contents of equations (3.4), (3.5), and (3.6) are more apparent when the H-function defined by relation (3.11) below is employed. Then the relationship between f and H follows from (3.4) and is

$$-f' = (H + r/\zeta)f.$$

Similar considerations apply for waves to the left. This leads to the following theorems:

Theorem 3.1 A transmission line is right distortionless if and only if

$$H' = H^2 + (\ell'/\ell)\, H, \tag{3.9}$$

$$(r/\ell)'\, H = 0, \tag{3.10}$$

with H(x) defined by

$$H = (g\zeta - r/\zeta - \zeta'/\zeta)/2. \tag{3.11}$$

The corresponding theorem for distortionless signalling to the left is

Theorem 3.2 A transmission line is left distortionless if and only if

$$\tilde{H}' = \tilde{H}^2 + (\ell'/\ell)\tilde{H}, \tag{3.12}$$

$$(r/\ell)'\, \tilde{H} = 0, \tag{3.13}$$

with $\tilde{H}(x)$ defined by

$$\tilde{H} = (-g\zeta + r/\zeta - \zeta'/\zeta)/2. \tag{3.14}$$

Suppose that a transmission line is both left and right distortionless. Then relations (3.9) and (3.12) both hold.
Subtracting these relations gives

$$(\tilde{H}-H)' = (\tilde{H}+H)(\tilde{H}-H) + (\ell'/\ell)\,(\tilde{H}-H). \tag{3.15}$$

Let

$$\beta = r/\zeta - g\zeta\ , \tag{3.16}$$

so $\tilde{H} - H = \beta$ and (3.15) becomes

$$\beta' = -\,(\zeta'/\zeta)\beta + (\ell'/\ell)\beta\ . \tag{3.17}$$

Integration of (3.17) gives $\beta\zeta/\ell$ = const. But $\zeta = \sqrt{\ell/c}$ so this relation becomes

$$\frac{r}{\ell} - \frac{g}{c} = \text{const.} \tag{3.18}$$

Corollary 3.1 A transmission line is left and right distortionless if:

$$H' = H^2 + (\ell'/\ell)H, \tag{3.19}$$

$$(r/\ell)' = 0\ , \tag{3.20}$$

$$(g/c)' = 0. \tag{3.21}$$

Proof. Relations (3.19) and (3.20) insure that the line is right distortionless. Relations (3.20) and (3.21) insure that relation (3.18) holds. But (3.18) implies (3.17) which implies (3.15). It is now evident that the conditions (3.12) and (3.13) of Theorem 3.2 hold and so the line is left distortionless.

Corollary 3.2 If r/ℓ is not constant on any interval the line is distortionless if and only if the characteristic impedance function is positive as in Theorem 2.1.

Corollary 3.3 *If r/ℓ is constant the line is right distortionless if and only if*

$$H' = H^2 + (\ell'/\ell)\, H, \tag{3.22}$$

where

$$H = 1/2\ (g\zeta - r/\zeta - \zeta'/\zeta). \tag{3.23}$$

Furthermore, the attenuation factor is

$$f(x) = \exp\left(-\int_0^x (H + r/\zeta)\, dx\right), \tag{3.24}$$

the delay is

$$\tau(x) = \int_0^x \sqrt{\ell c}\, dx, \tag{3.25}$$

and the characteristic impedance function is

$$Z = \frac{s + r/\ell}{s + (H + r/\zeta)/\tau'}\, \zeta, \tag{3.26}$$

where $s = 2\pi i\nu$ and ν is the frequency.
A similar Corollary for left distortionless transmission lines may be obtained.

4. *Applications*. The number of distortionless lines are nondenumerable. Several examples will be discussed.

Unidirectional lines. A unidirectional line satisfies either Theorem 3.1 or Theorem 3.2 and violates condition (3.18). Let

$$g = 0, \ell = 1,\ r = \zeta = \exp(-x).$$

It is seen that $H = (g\zeta - r/\zeta - \zeta'/\zeta)/2 = 0$. Thus by Theorem 3.1 the line is right distortionless. Relation (3.18) is not satisfied hence left distortionless waves are not possible.

Lines with the product ℓc = constant are characterized by a linear delay. This follows from relations (3.7) and (3.8). In addition to the unidirectional property, the example has a nonlinear delay time $\tau = \exp(x) - 1$. It is possible that a line which is both right and left distortionless has a nonlinear delay time.

The *general exponential line* is defined by the following relations among the parameters:

$$\underline{r} = \underline{\ell} = \underline{c}^{-1} = \underline{g}^{-1} = \exp\left[-\int_0^x \alpha(x)dx\right], \tag{4.1}$$

where $\underline{r}(x) = r/r_o$, $r(0) = r_o$ and similarly for $\underline{\ell}$, $\underline{c}$, and $\underline{g}$. Here $\alpha(x)$ is a function which characterizes the taper.

Corollary 4.1 *A general exponential line is right and left distortionless if $\alpha(x)$ satisfies*

$$(\alpha'/2) + (\alpha/2) = (\beta_o/2)^2, \tag{4.2}$$

with

$$\beta_o = r_o/\zeta_o - g_o\zeta_o . \tag{4.3}$$

Proof. Consider Corollary 3.1. By hypothesis, relations (3.20) and (3.21) are satisfied. From relations (2.3), (4.1) and (4.3), defintion (3.11) gives

$$H = (\alpha/2) - (\beta_o/2). \tag{4.4}$$

Equation (4.6.2) than follows from (3.19). Thus when (4.2) is satisfied so is Corollary 3.1. This completes the proof.

Equation (4.2) has the general solution

$$\alpha(x) = \beta_o \frac{e^{\beta_o x/2} - A\,e^{-\beta_o x/2}}{e^{\beta_o x/2} + A\,e^{-\beta_o x/2}} \ , \tag{4.5}$$

where A is an arbitrary real number. Special instances of the general proportional line are the exponential line due to Ghosh ($A = 0$ and $A = \infty$) and the Heaviside line ($\beta_o = 0$). Other interesting solutions are the hyperbolic lines characterized by $A = 1$ and $A = \exp(-\beta_o a)$, with $a > 0$.

In honor of Heaviside, who pioneered distortionless transmission lines, the H- function introduced in Section 3 is termed the Heavisidian.

REFERENCES

[1] R. Courant, "Hyperbolic Partial Differential Equations and Applications," E. F. Beckenbach (ed.), Modern Mathematics for the Engineer, McGraw-Hill, 1956, Chap. 5, pp. 92-109.

[2] O. Heaviside, "Electrical papers," Electrician, vol.2, p. 125 June 1887.

[3] S. K. Ghosh, "Responses in a non-uniform electrical transmission line," J. Inst. Eng. (India), Elec. Eng. Div., vol. 51, no. 12, Pt. E 26, pp. 329-330, August 1971.

[4] C. R. Burrows, "The Exponential Transmission Line," Bell System Technical Journal, vol. 17, pp. 555-573, October 1938.

[5] F. Bolinder, "Fourier transforms in the theory of inhomogeneous transmission lines," Proc. IRE, vol. 38, pg. 1354, November 1950.

[6] R. E. Collin, "The optimum tapered transmission line matching section," Proc. IRE, vol. 44, pp. 539-548, April 1956.

[7] Staff, Bell Telephone Labs., Transmission Systems for Communications, Bell Telephone Labs., 1964.

[8] O. K. Mawardi, "Generalized Solutions of Webster's Horn Theory," J. Acous. Soc. of America, vol. 21, July 1949, pp. 323-330.

[9] R. J. Duffin, "Distributed and lumped networks," Jour. of Math, and Mech., vol. 8, pp. 793-826, 1959.

[10] R. J. Duffin, "The extremal length of a network," Jour. of Math. Anal. and Appl., vol. 5, pp. 200-215, 1962.

[11] J. R. Pierce, "A note on the transmission line equation in terms of impedance," Bell System Technical Journal, vol. 22, 1943, pp.263-265.

ADDITIONAL TYPES OF MUTATORS AND ACTIVE RC SYNTHESIS USING MUTATORS

TADAO MURATA
Department of Information Engineering
University of Illinois at Chicago Circle
Chicago, Illinois 60680

ABSTRACT

This paper adds two types of L-R and C-R mutators to Chua's mutator and points out that some types of mutators have different names in the literature of active network synthesis. Also, typical examples of active RC synthesis using an L-R mutator or C-R mutator, which yields low-sensitivity values, are included.

I. INTRODUCTION

In 1968, L. O. Chua [1] introduced a new active linear two-port network called the mutator, as a device for transforming one type of nonlinear network element into another type. He gave two types of mutators for each of the following three classes of mutators: an L-R mutator, a C-R mutator and an L-C mutator.

In the following, the symbol T(p) denotes the transmission (ABCD) matrix as well as a mutator indicated by a subscript and p denotes the complex frequency variable. An L-R mutator is a device which transforms a resistor or conductor into an inductor, and types 1 and 2 are defined in the frequency domain, respectively, by

$$T_{LR1}(p) = \begin{bmatrix} p & 0 \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad T_{LR2}(p) = \begin{bmatrix} 0 & p \\ 1 & 0 \end{bmatrix}.$$

A C-R mutator is a device which transforms a resistor or conductor into a capacitor, and types 1 and 2 are defined, respectively, by

$$T_{CR1}(p) = \begin{bmatrix} 1 & 0 \\ 0 & p \end{bmatrix} \quad \text{and} \quad T_{CR2}(p) = \begin{bmatrix} 0 & 1 \\ p & 0 \end{bmatrix}.$$

An L-C mutator is a device which transforms a capacitor into an inductor, and types 1 and 2 are defined, respectively, by

$$T_{LC1}(p) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad \text{and} \quad T_{LC2}(p) = \begin{bmatrix} p & 0 \\ 0 & 1/p \end{bmatrix}.$$

This paper introduces two additional types of L-R mutators and C-R mutators, and points out that some types of mutators have different names in the literature of linear active network synthesis. In Section IV are presented two typical synthesis examples for a biquadratic transfer function using RC two-ports and an L-R mutator or C-R mutator. The active RC synthesis using an L-R or C-R mutator by the present method yields favorable results on the sensitivities of the pole-pair Q and ω_o, which are mostly of the order 0.5 or less.

II. ADDITIONAL TYPES OF L-R AND C-R MUTATORS

The input impedance $Z_{11}(p)$ of a two-port network described by the ABCD parameters and terminated by a resistor R is given by

$$Z_{11}(p) = \frac{AR + B}{CR + D} \tag{1}$$

or

$$Z_{11}(p) = \frac{A}{D} R \qquad \text{if } B = C = 0, \tag{2}$$

or

$$Z_{11}(p) = \frac{B}{C}\frac{1}{R} \qquad \text{if } A = D = 0. \tag{3}$$

Thus, an L-R mutator can be defined by A/D = p with B = C = 0, or B/C = p with A = D = 0. This suggests the following type 3 and 4 L-R mutators, in addition to Chua's $T_{LR1}(p)$ and $T_{LR2}(p)$:

$$T_{LR3}(p) = \begin{bmatrix} 1 & 0 \\ 0 & 1/p \end{bmatrix} \quad \text{and} \quad T_{LR4}(p) = \begin{bmatrix} 0 & 1 \\ 1/p & 0 \end{bmatrix}.$$

Similarly, a C-R mutator can be defined by A/D = 1/p with B = C = 0, or B/C = 1/p with A = D = 0, which is satisfied by $T_{CR2}(p)$ or $T_{CR2}(p)$ in the previous section as well as the following type 3 and 4 C-R mutators:

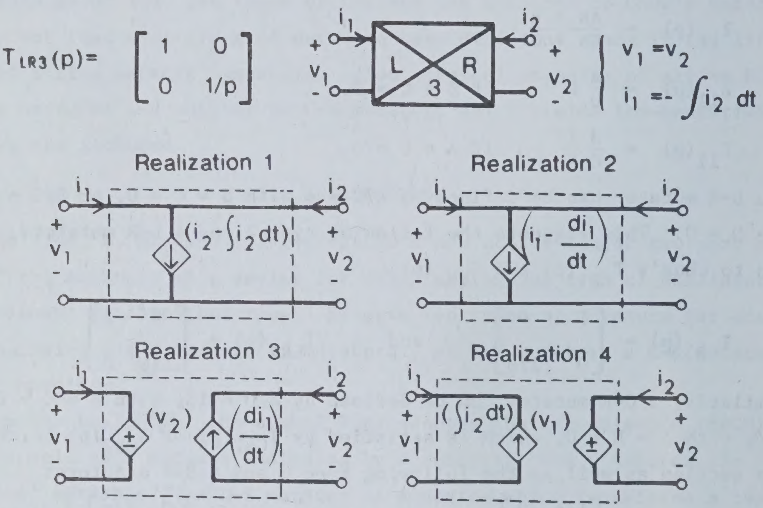
$$T_{CR3}(p) = \begin{bmatrix} 1/p & 0 \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad T_{CR4}(p) = \begin{bmatrix} 0 & 1/p \\ 1 & 0 \end{bmatrix}.$$

As was given in Table I of Chua's paper [1], it is easy to find the basic realizations using controlled sources of these new types of L-R and C-R mutators, which are shown in Figs. 1 and 2.

III. "MUTATORS" IN THE LITERATURE OF ACTIVE NETWORK SYNTHESIS

Linear active two-port networks characterized by the following ABCD parameters: A/D = K(p) with B = C = 0 and B/C = G(p) with A = D = 0 are called the immittance converter and the immittance inverter, respectively, in the literature of active network synthesis [2]. Thus, $T_{LR1}(p)$, $T_{LR3}(p)$, $T_{CR1}(p)$, $T_{CR3}(p)$, and $T_{LC2}(p)$ described in the previous sections evolve from the immittance converter, and $T_{LR2}(p)$, $T_{LR4}(p)$, $T_{CR2}(p)$, $T_{CR4}(p)$, and $T_{LC1}(p)$ described in the previous sections evolve from the immittance inverter.

A type 1 L-C mutator $T_{LC1}(p)$ is known as the gyrator and its use in active inductorless filter synthesis is so well-known that there is no need to mention it further here. $T_{LR3}(p)$ and $T_{CR1}(p)$ can be obtained as special

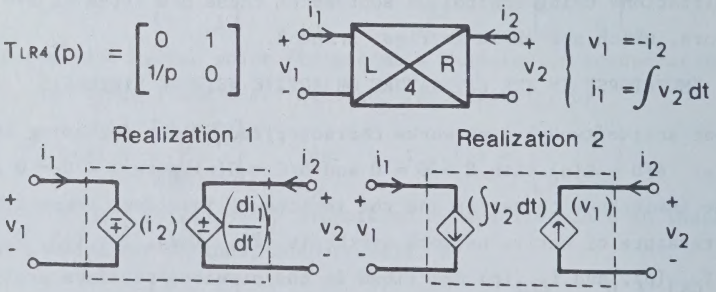


$$T_{LR3}(p) = \begin{bmatrix} 1 & 0 \\ 0 & 1/p \end{bmatrix}$$

$$\left\{ \begin{array}{l} v_1 = v_2 \\ i_1 = -\int i_2\, dt \end{array} \right.$$

i1
i2
+
v1
-
L
R
3
+
v2
-
Realization 1
(i2-∫i2dt)
Realization 2
(i1-di1/dt)
Realization 3
(v2)
±
(di1/dt)
Realization 4
(∫i2dt)
(v1)
±

(a) Type 3 L-R Mutator



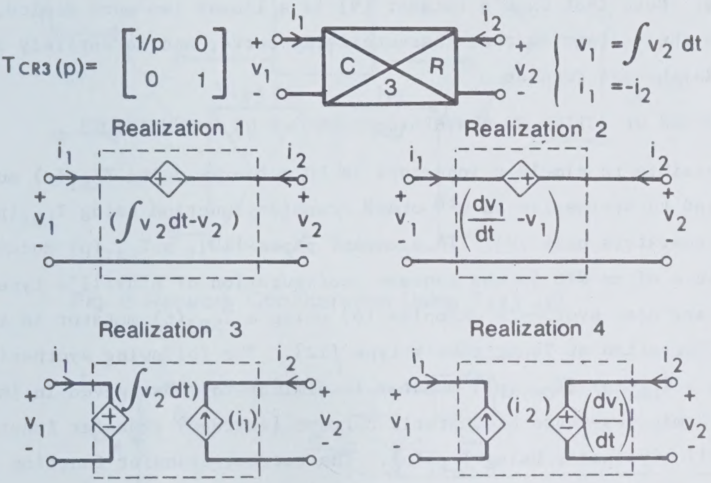
$$T_{LR4}(p) = \begin{bmatrix} 0 & 1 \\ 1/p & 0 \end{bmatrix}$$

$$\left\{ \begin{array}{l} v_1 = -i_2 \\ i_1 = \int v_2\, dt \end{array} \right.$$

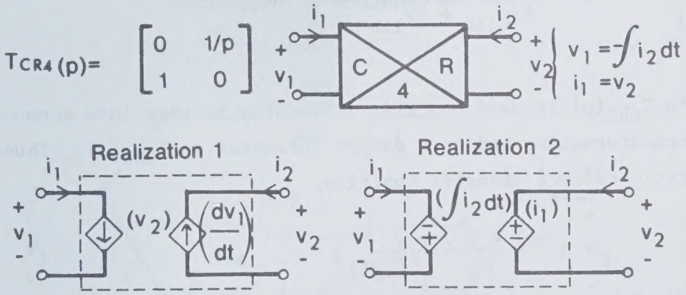
i1
i2
+
v1
-
L
R
4
+
v2
-
Realization 1
∓
(i2)
±
(di1/dt)
Realization 2
(∫v2dt)
(v1)

(b) Type 4 L-R MUTATOR

Fig. 1


TCR3(p)= [1/p 0; 0 1]
i1
i2
+ v1 -
C
3
R
+ v2 -
v1 = ∫v2 dt
i1 = -i2
Realization 1
(∫v2dt-v2)
Realization 2
(dv1/dt -v1)
Realization 3
(∫v2dt)
(i1)
Realization 4
(i2)
(dv1/dt)

(a) Type 3 C-R Mutator


TCR4(p)= [0 1/p; 1 0]
i1
i2
+ v1 -
C
4
R
+ v2 -
v1 = -∫i2dt
i1 = v2
Realization 1
(v2)
(dv1/dt)
Realization 2
(∫i2dt)
(i1)

(b) Type C-R Mutator

Fig. 2

cases of the Gorski-Popiel's positive immittance converter (PIC) [3]. Active RC synthesis using PIC realizes a low-sensitivity property [4] and a PIC may be a good alternative to the gyrator [5]. $T_{LR1}(p)$ or $T_{CR3}(p)$ [$T_{LR3}(p)$ or $T_{CR1}(p)$] is called a voltage [current] rotating-type rotator by Hayahara [6, 7, 8] in the sense that the output voltage [current] is rotated in phase by 90 degrees. Note that Chua's rotator [9] is a linear two-port device, which rotates the voltage-current characteristic curve, and is entirely different from Hayahara's rotator.

IV. EXAMPLES OF ACTIVE RC SYNTHESIS USING L-R OR C-R MUTATORS

It is possible to simulate inductors in LC networks using $T_{CR1}(p)$ mutators [3], and to synthesize an n^{th} order transfer function using $T_{LR1}(p)$ mutators and resistors only [8]. In a recent paper [10], a $T_{LR1}(p)$ mutator is used in place of an NIC in the cascade configuration of Linvill's type [11]. There are also synthesis examples [6] using a $T_{LR3}(p)$ mutator in the parallel configuration of Yanagisawa's type [12]. The following synthesis incorporating a $T_{LR2}(p)$ [$T_{CR1}(p)$] mutator is similar to those given in [6], and realizes a minimum-phase biquadratic current [voltage] transfer function.

Example 1: Synthesis Using $T_{LR2}(p)$. The current transfer function of the network shown in Fig. 3 can be written in terms of the y-parameters of the RC two-ports in the form,

$$\left.\frac{-I_2}{I_1}\right|_{V_2 = 0} = -\frac{ky_{21a}/y_{11a} + sy_{21b}}{k/y_{11a} + sy_{11b}}, \tag{4}$$

where $p = s/k$ in $T_{LR2}(p)$ is used and k is introduced to take into account the frequency transformation and/or a design parameter in $T_{LR2}(p)$. Thus, given a biquadratic current transfer function,

$$\begin{aligned}\left.\frac{-I_2}{I_1}\right|_{V_2 = 0} &= H\frac{\alpha s^2 + \beta s + \gamma}{s^2 + \frac{\omega_o}{Q}s + \omega_o^2} \\ &= \frac{H\gamma + s(H\alpha s + H\beta)}{\omega_o^2 + s(s + \frac{\omega_o}{Q})}\end{aligned} \tag{5}$$

the following associations can be made by comparing (5) with (4),

$$y_{11a} = k/\omega_o^2 \quad \text{and} \quad -y_{21a} = H\gamma/\omega_o^2\,, \tag{6}$$

$$y_{11b} = s + \omega_o/Q \quad \text{and} \quad -y_{21b} = H\alpha s + H\beta. \tag{7}$$

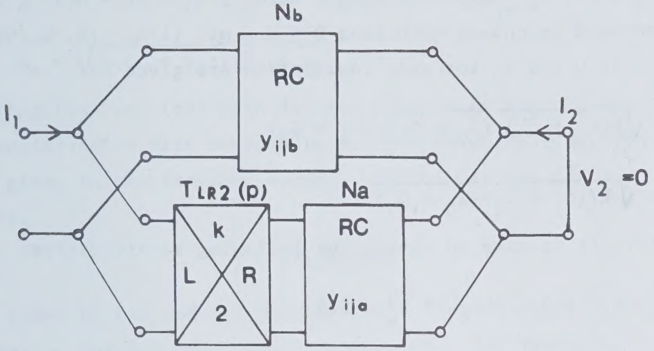

Nb
RC
y_{ijb}
I_1
I_2
V_2 =0
T_{LR2} (p)
k
L
R
2
Na
RC
y_{ija}

Fig. 3 Network Configuration Using T_{LR2} (p)

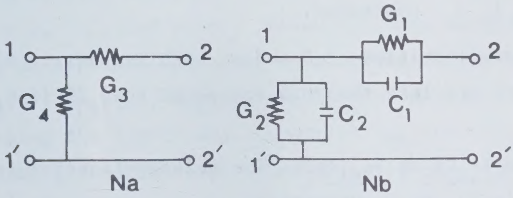

1
2
G_3
G_4
1´
2´
Na
G_1
1
2
C_1
G_2
C_2
1´
2´
Nb

Fig. 4 RC 2-ports for Example 1 (Ports 1 and 2 are exchanged for Example 2)

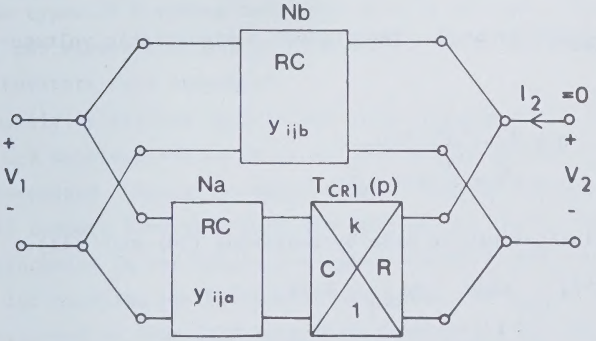

Nb
RC
y_{ijb}
+
V_1
-
I_2 =0
+
V_2
-
Na
RC
y_{ija}
T_{CR1} (p)
k
C
R
1

Fig. 5 Network Configuration Using T_{CR1} (p)

From (6) and (7), the two companion RC two-ports N_a and N_b can be synthesized as shown in Fig. 4, where the element values are given by $C_1 = H\alpha$, $C_2 = 1 - H\alpha$, $G_1 = H\beta$, $G_2 = \omega_o/Q - H\beta$, $G_3 = H\gamma/\omega_o^2$, $G_4 = (k - H\gamma)/\omega_o^2$, and the scaling factor H is chosen such that $0 < H \le \min\,[1/\alpha,\ \gamma/k,\ \omega_o/Q\beta]$.

The pole-pair Q and ω_o for this realization are given by

$$Q = \sqrt{k(C_1 + C_2)/(G_3 + G_4)}\Big/(G_1 + G_2) \tag{8}$$

and

$$\omega_o = \sqrt{k/(C_1 + C_2)(G_3 + G_4)}\ . \tag{9}$$

From (8) and (9), it is easy to obtain the following sensitivities:

$$S_{G_1}^{\omega_o} = S_{G_2}^{\omega_o} = 0, \qquad S_k^{\omega_o} = S_k^{Q} = 0.5$$

$$-S_{G_1}^{Q} = -S_{G_2}^{Q} \begin{cases} = 0.5 & \text{if } G_1 = G_2 \\ < 1 & \text{otherwise.} \end{cases}$$

The magnitudes of the sensitivities of Q and ω_o with respect to all the other network parameters are less than 0.5 and equal to 0.25 if $C_1 = C_2$ and $G_3 = G_4$.

Example 2: Synthesis Using $T_{CR1}(p)$. The voltage-transfer function of the network shown in Fig. 5 can be written as

$$\left.\frac{V_2}{V_1}\right|_{I_2 = 0} = -\frac{ky_{21a} + sy_{21b}}{ky_{22a} + sy_{22b}}, \tag{10}$$

where $p = s/k$ in $T_{CR1}(p)$ is used. Thus, given a biquadratic voltage-transfer function

$$\left.\frac{V_2}{V_1}\right|_{I_2 = 0} = H\,\frac{\alpha s^2 + \beta s + \gamma}{s^2 + \frac{\omega_o}{Q}s + \omega_o^2}\ , \tag{11}$$

the following associations can be made by comparing (10) with (11):

$$y_{22a} = \omega_o^2/k \quad \text{and} \quad -y_{21a} = H\gamma/k \tag{12}$$

$$y_{22b} = s + \omega_o/Q \quad \text{and} \quad -y_{21b} = H\alpha s + H\beta. \tag{13}$$

The companion RC two-ports realizing (12) and (13) are the N_a and N_b two-ports shown in Fig. 4 where ports 1 and 2 are exchanged. The element values are given by $C_1 = H\alpha$, $C_2 = 1 - H\alpha$, $G_1 = H\beta$, $G_2 = \omega_o/Q - H\beta$, $G_3 = H\gamma/k$, and $G_4 = (\omega_o^2 - H\gamma)/k$, where $0 < H \le \min\,[1/\alpha,\ \omega_o/Q\beta,\ \omega_o^2/\gamma]$.

The pole-pair Q and ω_o for this realization are given by

$$Q = \sqrt{(C_1 + C_2)(G_3 + G_4)/k} \Big/ (G_1 + G_2) \tag{14}$$

and $$\omega_o = \sqrt{k(G_3 + G_4)/(C_1 + C_2)} \; . \tag{15}$$

By comparing (14) and (15) with (8) and (9), it is easy to see that the Q and ω_o sensitivities with respect to all the network parameters are the same as those given in the previous example, except for the opposite sign in some parameters.

V. CONCLUSION

Two types of L-R and C-R mutators have been added to Chua's mutator. Theoretically, further additions are possible. For example, infinitely many L-R mutators could be defined by $B = C = 0$, $A = p^{n+1}$ and $D = p^n$, where $n = 0, \pm 1, \pm 2, \ldots$, or by $A = D = 0$, $B = p^{n+1}$ and $C = p^n$, where $n = 0, \pm 1, \pm 2, \ldots$, and so on. However, these mutators can be generated by cascading an appropriate number of mutators mentioned earlier. For that matter, $T_{LR3}(p)$, $T_{LR4}(p)$, $T_{CR3}(p)$ and $T_{CR4}(p)$ mutators could have obtained by interchanging the inputs and outputs of $T_{CR1}(p)$, $T_{CR2}(p)$, $T_{LR1}(p)$ and $T_{LR2}(p)$ mutators, respectively, since $T_{LR3}(p) = T^{-1}_{CR1}(p)$, $T_{LR4}(p) = T^{-1}_{CR2}(p)$, $T_{CR3}(p) = T^{-1}_{LR1}(p)$ and $T_{CR4}(p) = T^{-1}_{LR2}(p)$, where T^{-1} denotes the inverse of a matrix T. In this sense, Chua's list of mutators is complete.

Next, a review of the literature on active network synthesis has shown that some types of mutators have been used as active network elements. Finally, two examples of active RC synthesis incorporating $T_{LR2}(p)$ and $T_{CR1}(p)$ mutators were presented.

Sensitivity studies in this and other papers [4, 5, 10] indicate that L-R and C-R mutators are at least as good as the gyrator as far as sensitivity is concerned. Among the many types of mutators mentioned in this paper, some will compare favorably with the gyrator for use in active inductorless filter synthesis in the future when good practical mutators become available.

As for existing realizations of mutators, the $T_{LR1}(p)$ and $T_{CR1}(p)$ mutators proposed by Chua [13] have been constructed for use in experiments in nonlinear network synthesis. Hayahara [6, 7] designed his "rotators" using transistor amplifiers, and the Gorski-Popiel's PIC [3] can be readily used for realizing $T_{CR1}(p)$ and $T_{LR3}(p)$ mutators. However, further work would be necessary to develop practical mutators to be used in active filters.

REFERENCES

[1] L. O. Chua, "Synthesis of new nonlinear network elements," Proc. of the IEEE, Vol. 56, no. 8, pp. 1325-1340, Aug., 1968.

[2] S. K. Mitra, "Analysis and Synthesis of Linear Active Networks," New York, John Wiley & Sons, Inc., 1969.

[3] J. Gorski-Popiel, "RC-active synthesis using positive immittance converters," Electronics Letters, Vol. 3, No. 8, pp. 381-382, Aug., 1967.

[4] L. T. Bruton and A. B. Haase, "Sensitivity of generalized immittance converter-embedded ladder structures," IEEE Trans. on CAS, Vol. CAS-21, No. 2, pp. 345-350, March, 1974.

[5] J. Gorski-Popiel, "Positive impedance converter - an alternative to the active gyrator," Proc. of the 11th Midwest Symposium on Circuit Theory, Univ. of Notre Dame, pp. 489-499, May, 1968.

[6] E. Hayahara, "Realization of active impedance rotator and its application to transmission networks," Elect. and Communications in Japan, Vol. 53-A, No. 4, pp. 8-14, 1970.

[7] E. Hayahara, "Synthesis of active impedance rotator," Elect. and Communications in Japan, Vol. 55-A, No. 7, 1972.

[8] E. Hayahara, "Resistor rotator transfer network synthesis," Elect. and Communications in Japan, Vol. 56-A, No. 12, 1973.

[9] L. O. Chua, "The rotator - a new network component," Proc. of the IEEE, Vol. 55, No. 9, pp. 1566-1576, Sept., 1967.

[10] S. C. Goss and T. Murata, "Active RC synthesis using a type 1 L-R mutator: Cascade realization," to appear in Proc. of the 17th Midwest Symposium on Circuits and Systems, Univ. of Kansas, Sept., 1974.

[11] J. G. Linvill, "RC active filters," Proc. of IRE, Vol. 42, pp. 555-564, March, 1954.

[12] T. Yanagisawa, "RC active networks using current inversion type negative-impedance converters," IRE Trans. on Circuit Theory, CT-4, pp. 140-144, Sept., 1957.

[13] L. O. Chua and D. J. Gibson, "User's manual for rotator, reflector, scalar, and mutator," Report TR-EE 71-17, Purdue Univ., June, 1971.

[14] S. C. Goss, "Mutators and their uses in active RC network synthesis," M.S. thesis in preparation, Dept. of Info. Engrg., Univ. of Ill. at Chicago Circle, 1974.

DESIGN OF ACTIVE RC NETWORKS BY NONLINEAR PROGRAMMING

NICHOLAS KYRIAKOPOULOS, GILBERT GARDUNO
Department of Electrical Engineering and Computer Science
The George Washington University
Washington, D.C. 20006

ABSTRACT

Distributed RC networks in conjunction with operational amplifiers have been used to realize rational transfer functions. In this paper the problem has been formulated as a nonlinear programming problem where the objective function is the square of the difference between the desired characteristic and the equivalent characteristic of the network. The parameter space is formed by the coefficients of the cut functions, and the feasible region is determined by the realizability conditions imposed upon the network. Cut functions for specific problem formulations have been obtained.

INTRODUCTION

The use of shaped electrodes in distributed RC networks has originally been proposed by Heizer [1]. By virtue of their construction and geometry these networks give rise to driving point and transfer parameters which, in general, are irrational functions. Heizer's five terminal network may be used to realize rational short circuit admittance parameters; such realization does not, however, provide for frequency selectivity.

In a related development [2], a distributed network in conjunction with an operational amplifier has been used to realize second order transfer functions. Figure 1 shows the configuration used to realized open circuit voltage transfer functions.

The design has successfully implemented second order transfer functions. Attempts to obtain higher order implementations through analytical techniques have been unsuccessful. In this paper the problem is formulated as a nonlinear programming problem and a sample solution is obtained.

PROBLEM FORMULATION

Let the capacitance per unit area from the resistive layer to each of the three electrodes be given by $C^{0}(x,y)$, $C^{(3)}(x,y)$, $C^{(4)}(x,y)$.

Let $C^{(3)}(x,y)$ and $C^{(4)}(x,y)$ be selected as finite trigonometric series of the form

$$C^{(4)}(x,y) = C \sum_{k=1}^{M} \alpha_k \sin \frac{k\pi x}{L} = Cf_4(x)$$

$$C^{(3)}(x,y) = C \sum_{n=1}^{N} \beta_n \sin \frac{n\pi x}{L} = Cf_3(x) \tag{1}$$

where L is the length of the distributed network and C is the total capacitance per unit area. The short circuit admittance parameters y_{13} and y_{14} become

$$y_{13} = - \frac{W\pi}{RL} \sum_{n=1}^{N} \frac{n\beta_n s}{s+n^2\lambda}$$

(2)

$$y_{14} = - \frac{W\pi}{RL} \sum_{k=1}^{M} \frac{k\alpha_k s}{s+k^2\lambda}$$

where W is the width of the network, R is the layer resistivity and $\lambda = \pi^2/L^2RC$.

The connection of Fig. 1 under the assumption of high amplifier gain K, high input impedance and low output impedance, results in a voltage transfer function

$$T(s) = - \frac{y_{14}}{y_{13}} = - \frac{\sum_{k=1}^{M} (k\alpha_k/ (s+k^2\lambda))}{\sum_{n=1}^{N} (n\beta_n/ (s+n^2\lambda))} = \frac{A(\alpha_k,s)}{B(\beta_n,s)} \quad (3)$$

The coefficients of the polynomials A and B are functions only of the coefficients of the trigonometric series of Eqs. 1. Theoretically, the given configuration can realize any order transfer function. The only realizability condition imposed on T(s) is that the sets of real numbers $\{\alpha_k | k=1, \ldots, M\}$, $\{\beta_n | n=1, \ldots, N\}$ be such that

$$Cf_3(x)>0 \ ; \quad Cf_4(x)>0 \ ; \quad 0<x< L \qquad (4)$$

in addition, the cuts that define the electrodes must not intersect. From stability considerations it is also required that $B(\beta_n,s)$ be a Hurwitz polynomial.

Given then a desirable transfer function of the form

$$T(s) = \frac{\sum_{i=o}^{M} m_i s^i}{\sum_{j=o}^{M} n_j s^j} \qquad (5)$$

the coefficients of $A(\alpha_k,s)$ and $B(\beta_n,s)$ can be expressed in terms of the m_i s and n_i s. Unfortunately, the sets $\{\alpha_k | k=1, \ldots, M\}$, $\{\beta_n | n=1, \ldots, N\}$ resulting from the coefficient matching technique quite often violate the realizability conditions (4). In an effort to bypass the difficulty, the problem has been formulated as a nonlinear programming problem. The desired characteristic is given as a set of points and the objective function is defined in terms of this characteristic and the equivalent characteristic of the network.

Consider a desired insertion loss $\psi(\omega)$. The network insertion loss is

$$\xi(\omega) = \left| \frac{A(\alpha_k, j\omega)}{B(\beta_n, j\omega)} \right| \tag{6}$$

The objective function then may be defined as

$$F = \{\psi(\omega) - \xi(\omega)\}^2 \tag{7}$$

The problem thus becomes one of minimizing F over the feasible region of the decision variables, in this case the parameters α_k, β_n. The constraints imposed upon the solution are given by Eq. (4). One then has the problem of minimizing a nonlinear objective function subject to nonlinear constraints. An efficient optimization algorithm has been implemented [3] and used for the solution of this problem.

For ease of presentation of the procedure consider a second order transfer characteristic. From Eq. (3), the coefficients of A and B are, respectively,

$$\begin{aligned} M_2 &= \alpha_1 + 2\alpha_2 + 3\alpha_3 \ ; & N_2 &= \beta_1 + 2\beta_2 + 3\beta_3 \\ M_1 &= (13\alpha_1 + 20\alpha_2 + 15\alpha_3)\lambda \ ; & N_1 &= (13\beta_1 + 20\beta_2 + 15\beta_3)\lambda \\ M_0 &= (36\alpha_1 + 18\alpha_2 + 12\alpha_3)\lambda^2 \ ; & N_0 &= (36\beta_1 + 18\beta_2 + 12\beta_3)\lambda^2 \end{aligned} \tag{9}$$

From Eqs. (6) and (9) one obtains

$$\xi(\omega) = \left[\frac{(M_0(\underline{\alpha}) - M_2(\underline{\alpha})\omega^2)^2 + M_1^2(\underline{\alpha})\omega^2}{(N_0(\underline{\beta}) - N_2(\underline{\beta})\omega^2)^2 + N_1^2(\underline{\beta})\omega^2} \right]^{1/2} \tag{10}$$

where $\underline{\alpha}$ and $\underline{\beta}$ are the parameter vectors. From Eqs. (4) the constraints become

$$\begin{aligned} &\sum_{n=1}^{3} \beta_n \sin \frac{n\pi x}{L} > 0 \ ; \quad 0<x<L \\ &\sum_{k=1}^{3} \alpha_k \sin \frac{k\pi x}{L} > 0 ; \quad 0<x<L \end{aligned} \tag{11}$$

Given then a set of discrete data points the problem is formulated as follows

$$\min_{(\underline{\alpha},\underline{\beta})} \sum_{i=1}^{N} \left\{ \psi(\omega_i) - \left[\frac{(M_0(\underline{\alpha}) - M_2(\underline{\alpha})\omega_i^2)^2 + M_1^2(\underline{\alpha})\omega_i^2}{(N_0(\underline{\beta}) - N_2(\underline{\beta})\omega_i^2)^2 + N_1^2(\underline{\beta})\omega_i^2} \right]^{1/2} \right\}^2 \tag{12}$$

subject to constraints (9) and (11).

EXAMPLE

A desired insertion loss characteristic is given in Fig. 2. Assume a low pass voltage transfer function

$$T(s) = \frac{M_0}{s^2 + N_1 s + N_0} \qquad (13)$$

which can realize the desired characteristic. Setting $M_0 = N_0$ we can reduce the number of constraints to the following set

$$32\beta_1 + 10\beta_2 + 4 > 0 \quad ; \quad 8\beta_1 + 10\beta_2 + 5 > 0 \qquad (14)$$

$$(32\beta_1 + 10\beta_2 + 4)\left\{\frac{\sin\frac{\pi x}{L}}{24} - \frac{\sin\frac{2\pi x}{L}}{30} + \frac{\sin\frac{3\pi x}{L}}{120}\right\} > 0$$

$$\beta_1 \sin\frac{\pi x}{L} + \beta_2 \sin\frac{2\pi x}{L} + (1 - \beta_1 + 2\beta_2)\sin\frac{3\pi x}{L} > 0$$

The objective function to be minimized becomes

$$\min_{(\beta_1,\, \beta_2)} \sum_{i=1}^{N} \left\{ \psi(\omega_i) - \frac{(32\beta_1 + 10\beta_2 + 4)\lambda^2}{\left\{[(32\beta_1+10\beta_2+4)\lambda^2-\omega_i^2]^2+(8\beta_1+10\beta_2+5)^2\omega_i^2\right\}^{1/2}} \right\} \qquad (15)$$

RESULTS AND CONCLUSIONS

Since the problem has been formulated as a nonlinear programming problem, the solution possesses local minima. A choice then of different initial parameter vectors will most likely yield different solutions. Thus, various sets of initial vectors should be tried in order that an optimal choice for the resulting network design be made. Figure 2, indicates one of the obtained designs with the resulting network response. The coefficients of the series are $\beta_1 = 408.5$, $\beta_2 = -307.9$.

The procedure previously outlined can be extended to any order transfer function. Difficulties encountered or foreseen by the present approach are centered around the reduction of the large number of constraints appearing in the straightforward formulation of the problem and the selection of initial vectors which place the objective function within the feasible region.

REFERENCES

[1] Heizer, K.W., "Analysis of Rectangular RC Distributed Circuits with Shaped Electrodes", IEEE Transactions on Circuit Theory, Vol. CT-16, No. 1, pp. 8-13, February 1969.

[2] Meltzer, A.C., N. Kyriakopoulos, J. Saunders, "Analysis and Design of Distributed RC Filters", Conf. Rec. Fourth Asilomar Conference on Circuits and Systems, pp. 366 - 370, Nov. 1970.

[3] Kotsakis, G., "Optimization of System Function by Nonlinear Programming", M.S. Thesis, The George Washington University, Washington, D.C., 1971.

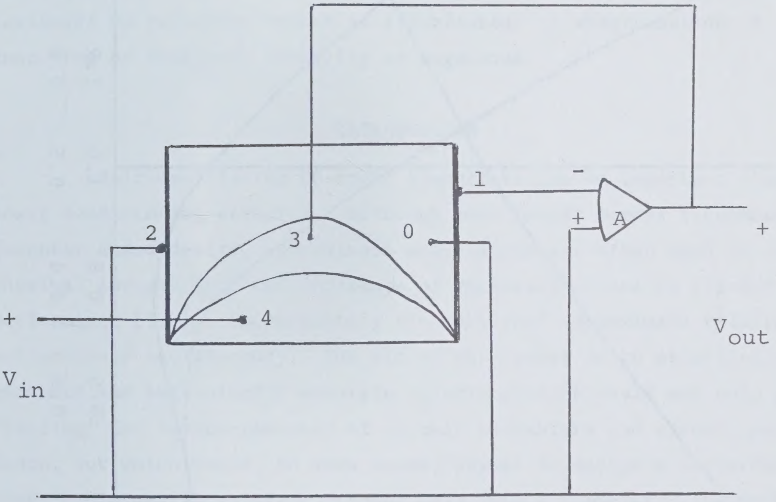


Figure 1

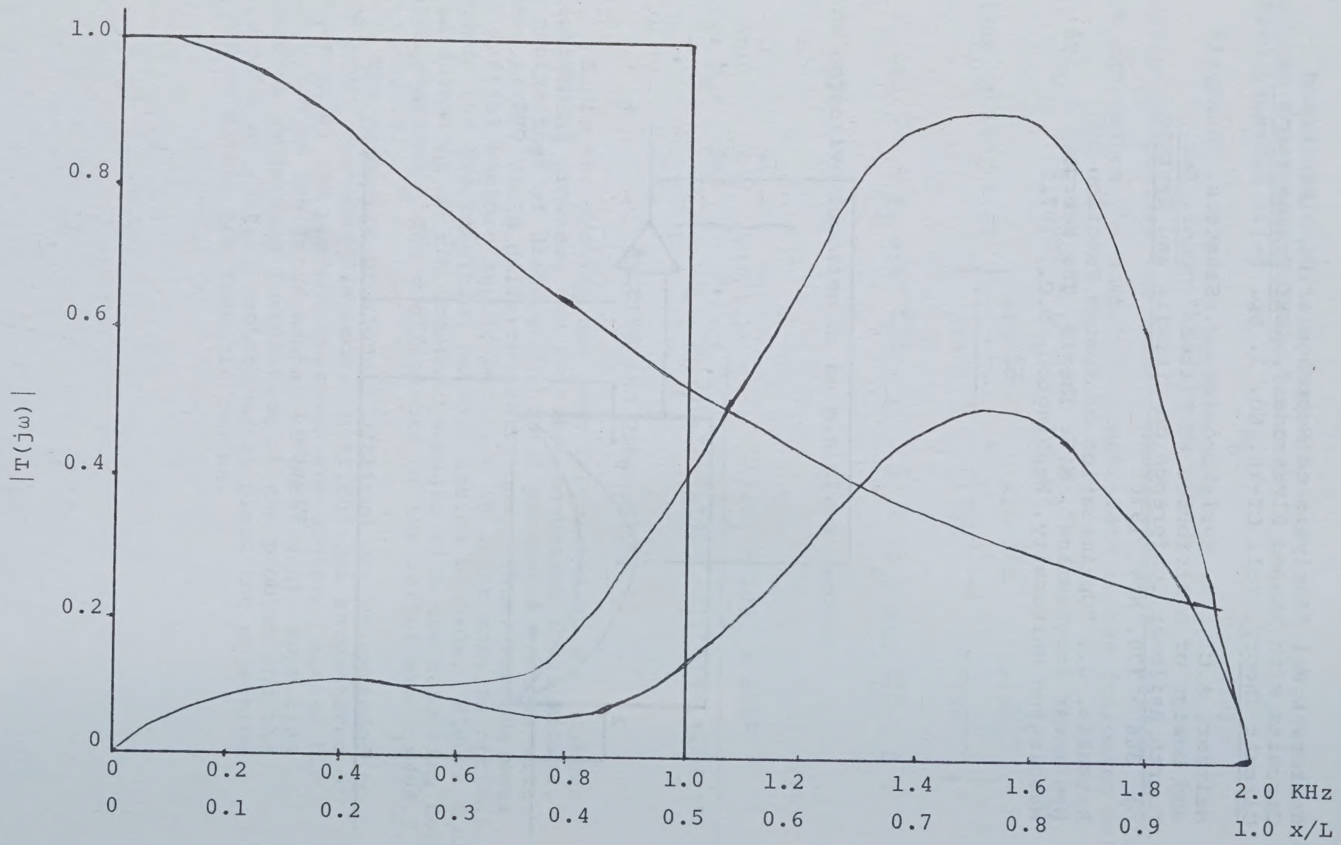

1.0
0.8
0.6
0.4
0.2
0
|T(jω)|
0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
1.8
2.0 KHz
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0 x/L

Figure 2

DESIGN EQUATIONS OF A HYSTERETIC DC-TO-DC CONVERTER

A. S. KISLOVSKI
Wylerringstrasse 1
3014 Bern
Switzerland

ABSTRACT

On the basis of the particularities which distinguish converters from other nonlinear oscillators, the exact differential equations of the most commonly used second-order linear filter are discussed. General characteristics of the solutions are established and a set of design relations giving the switching frequency, duty ratio and the bounds of the output voltage as explicit functions of circuit parameters is derived. Thus, the basis for understanding of the dependence of circuit characteristics on parameter values is established. A simple method of improving of frequency stability is suggested.

INTRODUCTION

Self-oscillating DC-to-DC converters are an important class of power conditioning circuits. Although most recent papers recommend a computer-aided design, approximate expressions are often used to give physical insight into the influence of parameter values on circuit performance [1,2]. Unfortunately the published approximate relations are not entirely satisfactory. The aim of this paper is to establish well grounded and sufficiently accurate relations which would not only give a "feeling" for interdependence of circuit parameters and circuit performance, but which would, in some cases, permit to design a converter with predictable characteristics without help from a computer. The works [4,5] intended for circuit designers do not contain such information.

We analyze only the case of the most often employed second-order filter. A block diagram of such a system is given in Fig. 1, the model of the linear filter in Fig. 2 and the corresponding idealized waveforms in Fig. 3. It is pointed out that, as a power-conditioning device, a converter must exhibit high efficiency which implies that the values of the dissipative elements r_L and r_C are small; output voltage of a realistic converter is supposed to remain within some relatively narrow limits so that the output ripple is small compared to the average value of the output voltage. These assumptions are of quite a general character and every properly designed converter will satisfy them.

GENERAL RELATIONS

Let t be equal to zero at the beginning of each considered interval. In that case,

$$L\,\frac{di}{dt} = \begin{cases} E - r_L i - u & 0 \le t \le t_{ON} \\ -r_L i - u & 0 \le t \le t_{OFF} \end{cases} . \qquad (1)$$

Since $r_L i$ is small and u is not much different from U_m and U_M, which are necessarily smaller than E, it follows that i is a monotonous function of time in each interval and that it assumes its amplitude values I_m and I_M exactly at the switching instants t_{OFF} and t_{ON}. If the system is oscillating, the output voltage ripple cannot be smaller than the hysteresis width, $U_M - U_m$. Under certain conditions it can, however, be greater and then the analytic expression for the corresponding waveform will have an extremum.

From (1) one obtains the ON interval:

$$\frac{du}{dt}\left(1 + \frac{r_C}{R}\right) = -\frac{1}{CR}E + \frac{1}{C}\left(1 + \frac{r_L}{R}\right)i + \left(r_C + \frac{L}{CR}\right)\frac{di}{dt} \qquad (2)$$

and, since $\frac{di}{dt}$ does not change sign, $\frac{du}{dt}$ remains positive during the whole ON interval if it is positive in the beginning of it. Hence the following necessary and sufficient condition for non-existence of a minimum of u

$$\frac{1}{C}\left(1 - \frac{r_C r_L C}{L}\right)I_m > \left(\frac{r_C}{L} + \frac{1}{CR}\right)U_m - \frac{r_C}{L}E . \qquad (3)$$

If a minimum exists, the lower bound is found to be:

$$u_{m\,min} = \frac{\frac{r_C}{L}E + \frac{1}{C}\left(1 - \frac{r_C r_L C}{L}\right)I_m}{\frac{r_C}{L} + \frac{1}{CR}} . \qquad (4)$$

An analogous reasoning yields the necessary and sufficient condition for non-existence of a maximum and, if it exists, its upper bound:

$$\frac{1}{C}\left(1 - \frac{r_C r_L C}{L}\right) I_M < \left(\frac{r_C}{L} + \frac{1}{CR}\right) U_M \tag{5}$$

$$u_{M\,MAX} = \frac{\frac{1}{C} - \frac{r_C r_L}{L}}{\frac{r_C}{L} + \frac{1}{CR}} I_M . \tag{6}$$

In a practical converter, the ripple is small compared to the average value U of the output voltage. Thus, one can write:

$$\int_0^{t_{ON}} u\, dt = U\, t_{ON} \quad , \quad \int_0^{t_{OFF}} u\, dt = U\, t_{OFF} \tag{7}$$

and considering that the losses are negligible,

$$(t_{ON} + t_{OFF}) \frac{U^2}{R} = \int_0^{t_{ON}} Ei\, dt \quad , \quad \frac{1}{2} L (I_M^2 - I_m^2) = \frac{U^2}{R} t_{OFF} . \tag{8}$$

Expressions (7), (8) combined with (1) yield:

$$t_{ON} = \frac{\left(1 + \frac{r_C}{R}\right)(U_M - U_m) L}{r_C \left\{E - \left[1 + \frac{r_L}{R + r_L} + \frac{L}{CR(R + r_L)} \cdot \frac{r_L}{r_C}\right] U\right\}} \tag{9}$$

$$t_{OFF} = \left(\frac{E}{U} \cdot \frac{R}{R + r_L} - 1\right) t_{ON} . \tag{10}$$

If $C \to \infty$, r_C, $r_L \to 0$, the last two expressions reduce to equations (38) and (39) of [1]. The amplitude values of inductor current i are found to be

$$I_M = A \frac{U}{R} + B \frac{E-U}{2L} t_{ON} \tag{11}$$

and

$$I_m = A \frac{U}{R} - B \frac{E-U}{2L} t_{ON} \tag{12}$$

where A and B are defined by the following expressions:

$$A = \frac{\frac{E}{U} \frac{R}{R + r_L} - 1}{\frac{E}{U} - 1 - \frac{r_L}{R + r_L}} \quad , \quad B = 1 - \frac{r_L}{R + r_L} \cdot \frac{U}{E-U} . \tag{13}$$

Thus, the switching frequency is determined explicitly and the bounds of output voltage ripple, if it is superior to the difference of threshold voltages, can be evaluated from the expressions derived above.

REDUCING FREQUENCY VARIATION

The switching frequency can be influenced either by varying the hysteresis width or by influencing one of the threshold voltages. A constant-frequency system with a separate feedback loop based on the first possibility has been reported [3]. Provided that r_C and r_L are small and C large, one concludes from (9) and (10) that the main reason for the frequency instability is the variation of the input voltage E.

Let us suppose that every ON interval is prolonged by a fixed delay τ. In that case, the period of oscillations is given by:

$$T = \frac{E}{U} \cdot \frac{R + r_C}{R + r_L} \left(\frac{L}{r_C} \cdot \frac{U_M - U_m}{E - \alpha U} + \frac{R}{R + r_C} \tau \right) \tag{14}$$

where

$$\alpha = 1 + \frac{r_L}{r_L + R} + \frac{L}{CR(R + r_L)} \cdot \frac{r_L}{r_C} . \tag{15}$$

If (14) is differentiated with respect to E and the result equalled to zero, then

$$\tau = \alpha \frac{L}{r_C} \frac{(U_M - U_m)U}{(E - \alpha U)^2} \left(1 + \frac{r_C}{R}\right) \tag{16}$$

and τ can be determined so that frequency variations be kept at minimum in the neighborhood of a certain value of the supply voltage E.

EXPERIMENTAL TESTS

A circuit with L = 3.8 mH, r_L = 3.6 Ohm, C = 290 F, R = 20 Ohm, r_C = 0.41 Ohm, $U_M - U_m$ = 35 mV, U = 5 V and a delay τ of 6.6 μs was constructed and tested. The chosen delay corresponds to $\frac{dT}{dE} = 0$ at E = 29.3 V. Calculated and measured values of T are represented in Fig. 4. For comparison, a period of uncompensated converter is plotted too. Considering a limited interval for the input voltage from 22 to 44 V, one concludes that the frequency variation can be reduced by a factor of more than 5 if this approximate compensation method is used.

CONCLUSION

Guidelines for a straightforward design of a realistic converter are established. Sufficient accuracy and predictable performance can be achieved without a computer-aided design. The derived equations show that the switching frequency of properly designed converter is practically independent of the load. A simple method of reducing of the frequency dependence on the input voltage is suggested. Calculated and measured switching frequency periods are compared.

REFERENCES

[1] I. M. H. Babaa, T. G. Wilson, and Y. Yu, "Analytic Solutions of Limit Cycles in a Feedback - Regulated Converter System with Hysteresis," IEEE Trans. Automatic Control, Vol. AC-13, October 1968.

[2] F. F. Judd and C. T. Chen, "Analysis and Optimal Design of Self-Oscillating DC-to-DC Converters," IEEE Trans. Circuit Theory, Vol. CT-18, November 1971.

[3] F. F. Judd and H. Wilhart, "Self-Oscillating Regulated DC-to-DC Converter," IEEE Trans. on Ind. Appl., November/December 1972.

[4] E. R. Hnatek, Design of Solid-State Power Supplies, Van Nostrand Rheinhold Co., 1971.

[5] S. W. Wagner, Stromversorgung elektronischer Schaltungen und Geräte, R. V. Decker's Verlag G. Schenk, Hamburg, 1964.

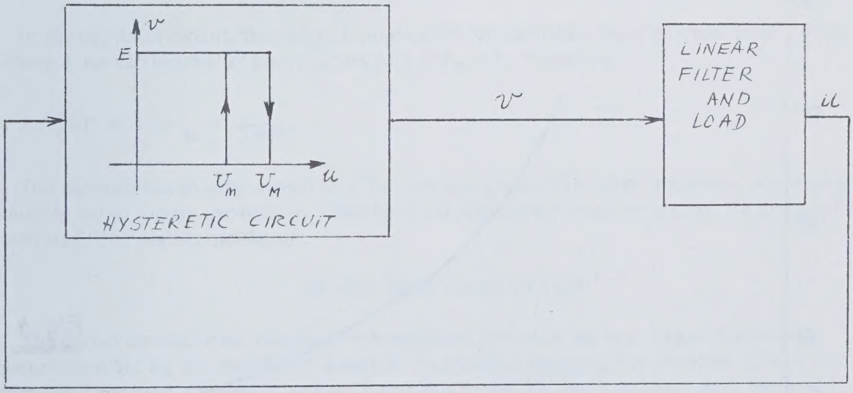


Fig 1

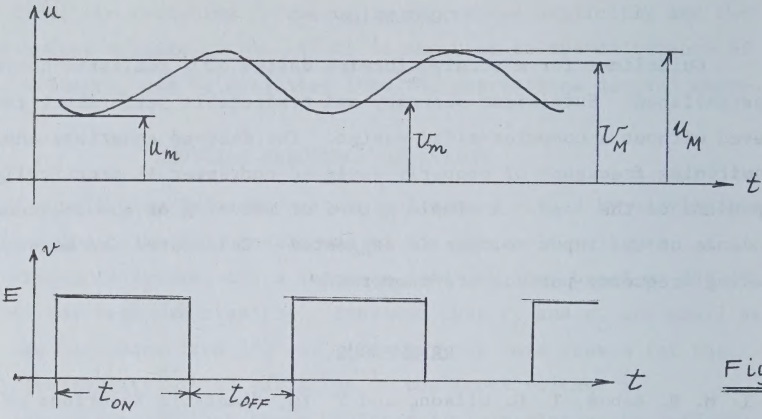
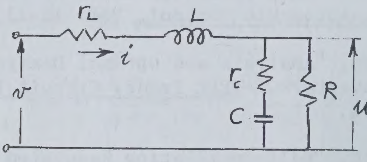

u
u_m
U_m
U_M
u_M
t
v
E
t_{ON}
t_{OFF}
t

Fig. 2

Fig. 3


r_L
L
i
v
r_c
C
R
u

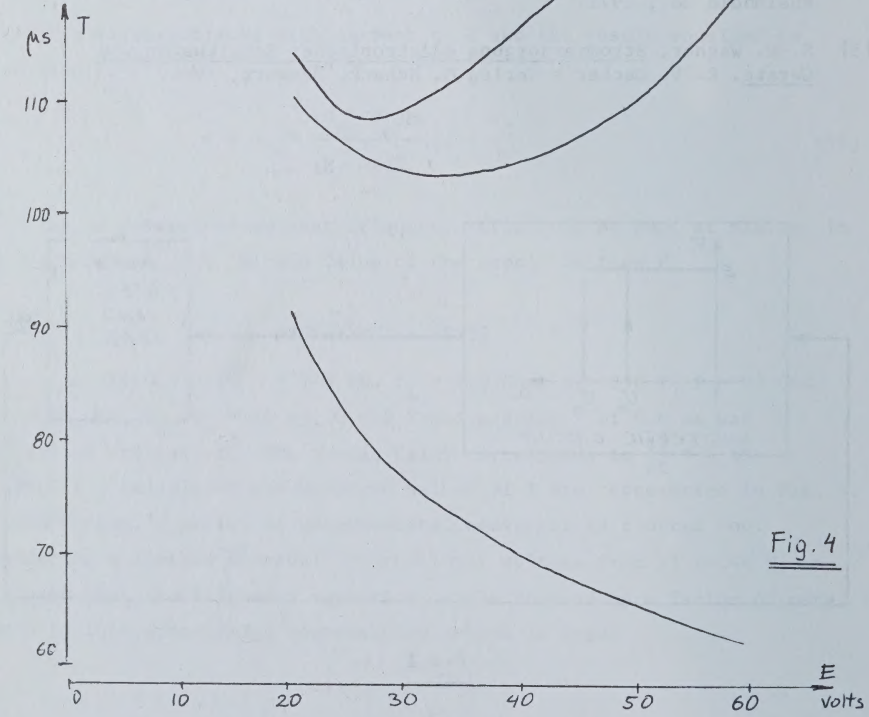


Fig. 4

THE WIEN BRIDGE OSCILLATOR REVISITED (OR, THE WIEN BRIDGE OSCILLATOR USED AS A KEYED OSCILLATOR OR A BANDPASS FILTER)

RICHARD E. HELD
Harris Corporation, Electronic Systems Division, Palm Bay, Florida

The Wien Bridge Oscillator is an extremely stable signal source over the range of from about 10 Hz to 1.0 MHz using standard low-cost components. This stability and accuracy can be used to advantage in keyed oscillators and bandpass filters by playing games with the feedback networks. This article presents the basic Wien Bridge Circuit and then the modifications to it to perform these functions.

BASIC CIRCUIT

Figure 1 shows the basic Wien Bridge Oscillator circuit. Resistors R_1 and R_2 attenuate the negative feedback to the 741 type operational amplifier setting its forward loop gain. If this gain is adjusted to the value such that its gain multiplied times the network attenuation tied to the + input is equal to unity, oscillation will occur. Also, the phase shift of the network tied to the + input must be 0^o at the same point to cause oscillation.

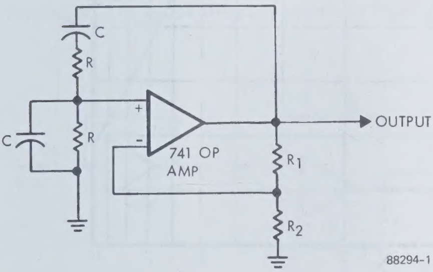


Figure 1. Basic Wien Bridge Oscillator

The network tied to the + feedback input is designed using equal resistors and equal capacitors. If this is the case, the transfer function of this network is:

$$\frac{E\ OUT}{E\ IN} = \frac{1}{3 + SCR + \frac{1}{SCR}}$$

If we replace the complex operator S with $j2\pi f$, set C and R both equal to 1, and then replace f with F/F_o, we can obtain the normalized plot of gain and phase shift of the network. Figure 2 is the plot of gain and phase shift of the network versus frequency.

In the oscillator circuit, the network phase shift for oscillation must be equal to 0^o. From Figure 2, we can see the 0^o point occurs with $F/F_o = 1$. Therefore,

$$2\pi f_o RC = 1 \text{ or } f_o = \frac{1}{2\pi RC} .$$

The network has an attenuation of 1/3 at the zero phase shift point, therefore, amplifier gain must be 3.0 to sustain oscillation. Therefore, the attenuation ratio for R_1 and R_2 in Figure 1 must be 1/3 to sustain oscillation.

KEYED TONE GENERATOR

The circuit can easily be modified for keyed tone generator service. Figure 3 shows the circuit modified by the addition of a switch. In the open position, the amplifier gain is 1.0. With this condition, oscillation will not occur due to insufficient loop gain. With the switch closed, the amplifier gain goes to 3.0. With this gain, the loop gain is unity and oscillation occurs.

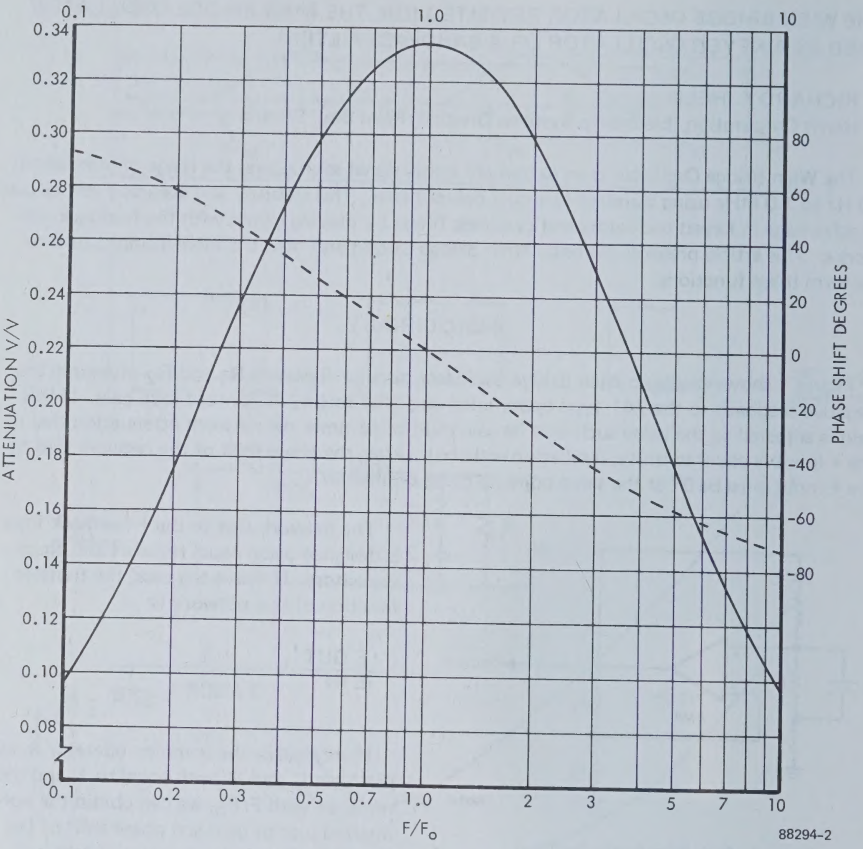


Figure 2. Network Gain and Phase Versus Frequency

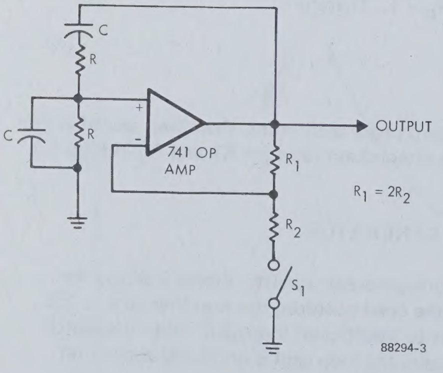


Figure 3. Keyed Tone Generator

The practical circuit using this concept is shown in Figure 4. The feedback gain is adjusted to give a loop gain slightly greater than unity, to guarantee oscillation and diodes are added to limit the gain once oscillation occurs and thus, control output amplitude. The mechanical switch is replaced with a transistor to allow direct interfacing with logic controls. More exotic circuitry using an FET transistor in place of the diodes and the junction transistor could be used if less distortion or more accurate output is desired.

This circuit is very attractive as it was used for a tone annunciator in a PA system. The key function is such that the tone starts at zero amplitude and builds up to stable output in about 4 cycles of the oscillator frequency. It decays to zero output within 1 cycle. As a result, there are no clicks and POPS that are the normal discontinuities associated with keyed oscillators. The tone sound thus, is pleasing to the ear.

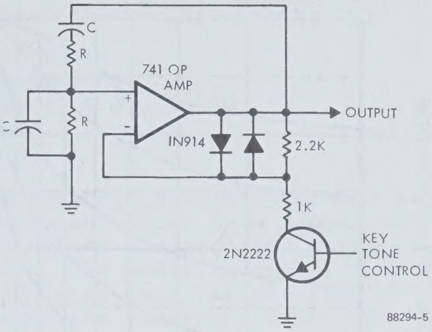

C
R
741 OP AMP
+
-
C
R
IN914
OUTPUT
2.2K
1K
2N2222
KEY TONE CONTROL
88294-5

Figure 4. Practical Keyed Tone Generator

BANDPASS FILTER

The circuit can be converted into a bandpass filter in which Q and center frequency can be independently selected. If the loop gain is adjusted to be less than unity, and the circuit arranged to allow for insertion of an external input, a bandpass filter results. Figure 5 shows two typical connections for the filter. Either connection yields the same result as long as the output impedance of the isolating amplifier is low with respect to the filter element impedances. In many cases, the isolating amplifier can be deleted, however, the source impedance from the external input should be low enough to not impact the network impedances R and C. If we look at the second amplifier configuration, the block diagram shown in Figure 6 results.

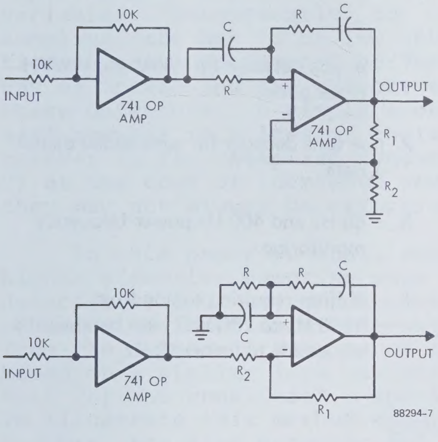

10K
10K
INPUT
741 OP AMP
C
R
R
C
+
-
OUTPUT
741 OP AMP
R1
R2
10K
10K
INPUT
741 OP AMP
R
R
C
C
R2
+
-
OUTPUT
R1
88294-7

Figure 5. Typical Filter Configurations

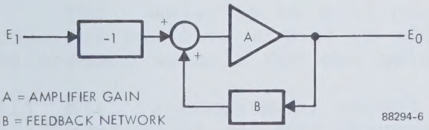

E1
-1
+
+
A
Eo
B
A = AMPLIFIER GAIN
B = FEEDBACK NETWORK
88294-6

Figure 6. Filter Block Diagram

Applying the basic closed loop equation of

$$E_o/E1 = \frac{A}{1+\beta A}$$

and recognizing A takes on values depending on the ratio

$$\frac{R_1+R_2}{R_2}.$$

The resulting descriptive equation is:

$$\text{Gain} = \frac{E_o}{E_1} = \frac{1}{1 - A\left(\frac{1}{3 + SRC + \frac{1}{SRC}}\right)}$$

If this equation is plotted with various values for A, the filter characteristics shown in Figure 7 result. Note in the figure the response is plotted in decibels with 0 dB at the center frequency. Figure 8 shows actual filter gain at the center frequency versus amplifier forward gain.

This filter configuration has been used in several different applications requiring moderately narrow bandwidths and center frequency tuning. Some of these applications have been:

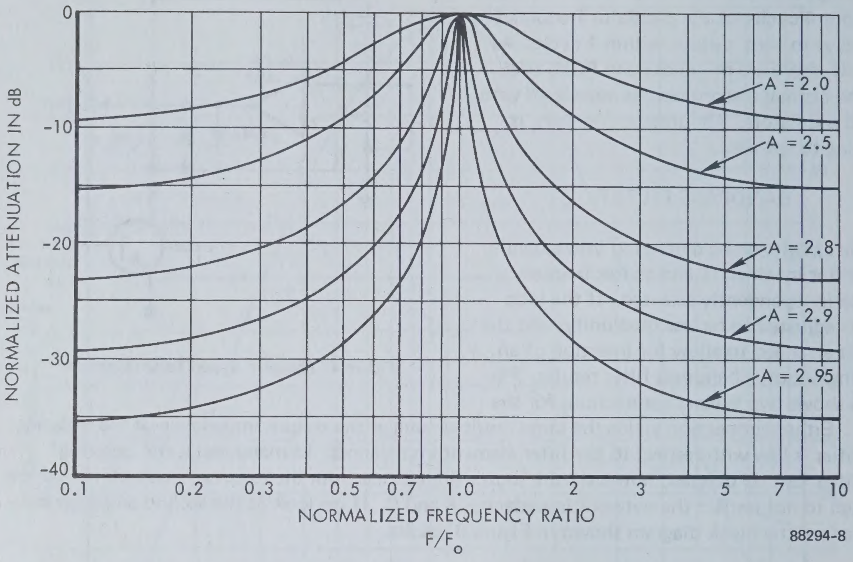


Figure 7. Filter Characteristics

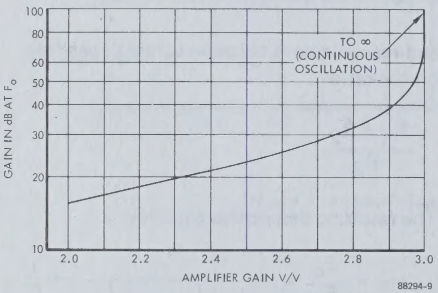


Figure 8. Gain at Center Frequency F

1. A tick detector for WWV and WWVH clock pulses
2. A tone decoder for tone coded digital data
3. 60 Hz and 400 Hz power frequency monitoring
4. A filter requiring tuning from 1900 Hz to 2300 Hz with bandwidths adjustable to below 10 Hz

CONCLUSIONS

The circuit, using the Wien Bridge Oscillator as a starting point, is very versatile. When Tone Coding and Decoding are required, or single resonance bandpass filtering is required, this circuit may prove to be quite useful. It certainly has been here at Harris Corporation.

DEPENDENT SAMPLING IN NON-ADAPTIVE NONPARAMETRIC DETECTORS

SALEEM A. KASSAM AND JOHN B. THOMAS
Department of Electrical Engineering
Princeton University, Princeton, N.J. 08540

ABSTRACT

A class of non-adaptive nonparametric detectors with distribution-free performance for dependent input data sequences is considered. The detectors in this class are modified versions of standard nonparametric detectors, the modifications consisting of an arrangement to group input data as multivariate samples, followed by a multivariate-to-univariate transformation. Performance analysis is initially based on the assumption that the multivariate samples are independent, and the results are then shown to be applicable under weaker conditions. The results indicate the effectiveness of the detectors in achieving higher efficiencies over those obtained by standard nonparametric detectors restricted to independent sampling.

I. INTRODUCTION

In establishing the constancy of the false-alarm rate of nonparametric detectors it is almost always assumed that the input data is a sequence of statistically independent random variables. Consequently, for continuous-time processes, the sampling rate has to be low enough to ensure that this assumption is met. To improve performance it is desirable to have a way of increasing the effective sampling rate at the inputs of these detectors. A simple increase in the sampling rate by itself results in a loss of their nonparametric performances, however [1,2]. Adaptive nonparametric detectors can be used [3-7] at the cost of increased complexity of implementation, and they may not always be effective.

In this paper we shall consider a method of achieving a higher effective sampling rate at the input of a nonparametric detector without using adaptive procedures. The method consists of using as input data independent multivariate samples derived from the dependent samples of the input process. An approach based on a similar idea has also been suggested by Woinsky and Kurz [8] who considered a specific nonparametric detection scheme. To illustrate this method we will consider two one-channel detectors, the sign detector and the Wilcoxon detector. The method, however, has quite general application.

II. FORMULATION AND METHOD

For simplicity we will consider the problem of detecting a constant positive signal in additive noise. The following will be assumed to hold for the noise process $N(t)$:

(i) $N(t)$ is zero-mean, second order, stationary, mean square continuous and has diagonally symmetric probability density functions of all finite orders.

The assumption of stationarity can be relaxed, as we will see.

Diagonal symmetry of the noise probability density functions is a valid assumption in all cases of interest.

(ii) There exists a finite maximum sampling rate $r_0=1/T$ such that for all rates $r \leq r_0$ the samples can be assumed to be almost independent, i.e. independent for the purpose of nonparametric detector performance analysis [4,9]. The Gauss-Markov process is an example of such a process, where sampling at a rate equal to the double-sided bandwidth gives rise to samples which can be assumed to be independent [9].

For exact results in what follows we will in fact assume that the normalized autocorrelation function $\rho_N(\tau)$ of the noise process vanishes outside of a finite interval $(-T,T)$, and that samples separated by an interval $\geq T$ are independent. A large class of processes satisfy these even more restrictive assumptions, examples being memoryless nonlinear transformations of Gaussian processes with autocorrelations of finite support, finite moving average processes, and the simple shot noise process with finite pulse width. We will also discuss later how the results are to be interpreted when these latter assumptions are not exactly satisfied.

A sampling rate r_0 thus gives a sequence of independent samples $X_1, X_2, \ldots X_N$, where $NT=t_0$, the observation interval. If the sampling rate is increased by a multiplicative factor γ, where $\gamma=1,2,\ldots$, we get a $(\gamma-1)$-dependent sequence of samples $Y_1, Y_2, \ldots, Y_{N\gamma}$. Consider grouping n consecutive Y samples as one multivariate sample, with $\gamma-1$ samples skipped between the groups. This gives a sequence of independent n-variate samples $\underline{Z}_1, \underline{Z}_2, \ldots, \underline{Z}_{N\gamma/n+\gamma-1}$ (assuming $N\gamma/n+\gamma-1$ is an integer), with

$$\underline{Z}_i = (Y_{(n+\gamma-1)(i-1)+1}, Y_{(n+\gamma-1)(i-1)+2}, \ldots, Y_{(n+\gamma-1)(i-1)+n}). \quad (1)$$

Let D_X be the nonparametric detector operating on the independent sequence $X_1, X_2, \ldots, X_N$ in some specific way. We will consider its modification D_Z which will operate on the sequence $\underline{Z}_1, \underline{Z}_2, \ldots, \underline{Z}_{N\gamma/n+\gamma-1}$. Denote by D_Y the adaptive threshold detector [4] based on D_X. A comparison of the performances in these three configurations will be made for the case of the sign detector and the Wilcoxon detector.

The detector D_Z will be restricted to operate, just as D_X, on $N\gamma/n+\gamma-1$ independent univariate data values derived by transforming each $\underline{Z}$ sample. Perhaps the simplest non-trivial transformation is the summing operation, i.e. the components of each $\underline{Z}$ sample are summed to obtain a single variable. We will restrict attention to this particular transformation; however, sums of nonlinear functions of the components of the $\underline{Z}$ samples also lead to interesting results.

III. PERFORMANCE

To compare the performances of two detectors we will use their asymptotic relative processing time (ARPT), obtained from their asymptotic relative efficiency (ARE). The ARE, which is a weak-signal large-sample efficiency measure, can be computed readily in our applications using the Pitman-Noether theorem [10].

The $\mathrm{ARPT}_{Z,X}$ of detector D_Z relative to D_X is defined by

$$\mathrm{ARPT}_{Z,X} = \mathrm{ARE}_{Z,X} \times [\gamma/(n+\gamma-1)] \tag{2}$$

where $\mathrm{ARE}_{Z,X}$ is the asymptotic relative efficiency of D_Z with respect to D_X. Thus ARPT is a comparison of the lengths of observation times required by the two detectors to achieve the same asymptotic performance.

(a) Modified Sign Detector

With the multivariate-to-univariate transformation being achieved through the summing operation we get a modified sign detector D_Z with test statistic

$$S_Z = [N\gamma/(n+\gamma-1)]^{-1} \sum_{i=1}^{N\gamma/(n+\gamma-1)} \mathrm{Sgn}(Z^i) \tag{3}$$

where

$$\mathrm{Sgn}(x) = \begin{cases} 1, & x>0 \\ 0, & x=0 \\ -1, & x<0 \end{cases}$$

and Z^i = sum of the n components of $\underline{Z}_i$.

Using the Pitman-Noether theorem, we find that, for the sign detector,

$$\mathrm{ARPT}_{Z,X} = n^2 [f_Z^2(0)/f_N^2(0)][\gamma/(n+\gamma-1)] \tag{4}$$

where $f_Z(z)$ is the common univariate probability density function of the Z^i under the hypothesis of no signal present, and $f_N(x)$ is the univariate noise probability density function.

If the n-variate noise probability density function is known then $f_Z(0)$ can be found, and Eq. (4) can be evaluated. In the case of Gaussian noise we can easily evaluate the above ARPT to find that

$$\mathrm{ARPT}_{Z,X} = n^2 [n+2 \sum_{i=1}^{\min(n,\gamma)} (n-i)\rho_N(iT/\gamma)]^{-1} [\gamma/(n+\gamma-1)]. \tag{5}$$

In the limit as n→∞ (increasing group size) for fixed γ we get

$$\lim_{n\to\infty} \mathrm{ARPT}_{Z,X} = \gamma[1+2 \sum_{i=1}^{\gamma-1} \rho_N(iT/\gamma)]^{-1}, \tag{6}$$

and if we let $\gamma\to\infty$ also we get

$$\lim_{\substack{\gamma\to\infty \\ n/\gamma\to\infty}} \mathrm{ARPT}_{Z,X} = (T/2)\left[\int_0^T \rho_N(\tau)d\tau\right]^{-1} . \tag{7}$$

It is easily shown that $\int_0^T \rho_N(\tau)d\tau \leq T/2$, so that, in the limiting case of high sampling rate and very large group size, we have $\mathrm{ARPT}_{Z,X} \geq 1$. For finite values and some common $\rho_N(\tau)$, the ARPT's are shown in Figs. 1 and 2. Tables I and II give corresponding numerical values of $\mathrm{ARPT}_{Y,X}$ for comparison. The equations for $\mathrm{ARPT}_{Y,X}$ can be derived by modifying those given in [4] to apply to our cases. Here D_Y is the adaptive threshold sign detector, with perfect adaptation being equivalent to exact knowledge of the variance of the test statistic.

For non-Gaussian noise processes, $f_Z(0)$ can be approximated for large enough n by the Gaussian probability density function evaluated at zero. Using this approximation we have

$$\mathrm{ARPT}_{Z,X} \approx n^2\left[n+2\sum_{i=1}^{\gamma-1}(n-i)\rho_N(iT/\gamma)\right]^{-1}\left[2\pi\sigma_N^2 f_N^2(0)\right]^{-1}\left[\gamma/(n+\gamma-1)\right]$$

$$\text{(large n)} \tag{8}$$

where σ_N^2 is the noise variance. For the specific case where the noise process has a univariate uniform probability density function we then obtain $\mathrm{ARPT}_{Z,X} \approx (6/\pi)$ [$\mathrm{ARPT}_{Z,X}$ for the corresponding Gaussian case]. On the other hand, for the double-exponential noise probability density function, $\mathrm{ARPT}_{Z,X} \approx (1/\pi)$ [$\mathrm{ARPT}_{Z,X}$ for the corresponding Gaussian case]. The approximations can be shown to be very good for moderate values of n.

Another model for non-Gaussian noise is the output of an odd, monotone zero-memory nonlinearity driven by Gaussian noise. It can be shown that, for the case n=2, the effectiveness of D_Z is decreased when the nonlinearity tends to spread out the tails of the Gaussian density, and vice-versa. This is in agreement with the results on non-Gaussian noise obtained above.

For the modified sign detector D_Z an odd pre-detection nonlinearity does not change the false-alarm probability α of the detector. The nonlinearity may be time-varying and the noise may be non-stationary as long as the assumptions of diagonal symmetry of the probability density functions and of the previously stated dependency structure of the process hold. For the case n=2, if the nonlinearity is monotone as well as odd, the power β of the detector is also preserved. But non-monotone odd nonlinearities could be used to increase the power of D_Z.

The overall performance of the modified sign detector D_Z relative to the Neyman-Pearson detector optimal for Gaussian noise is very good for many types of autocorrelation functions. This latter detector is designed either for independent samples or for a particular form of autocorrelation function of the noise process. Therefore in comparing D_Z with it we restrict the

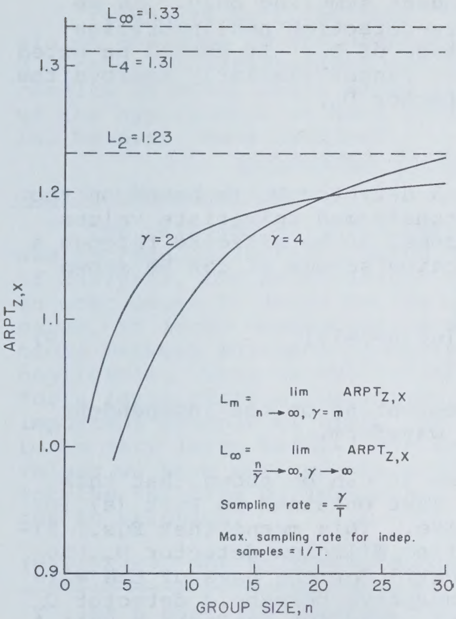


Fig. 1 $\mathrm{ARPT}_{Z,X}$ for sign and Wilcoxon detectors, case (i)

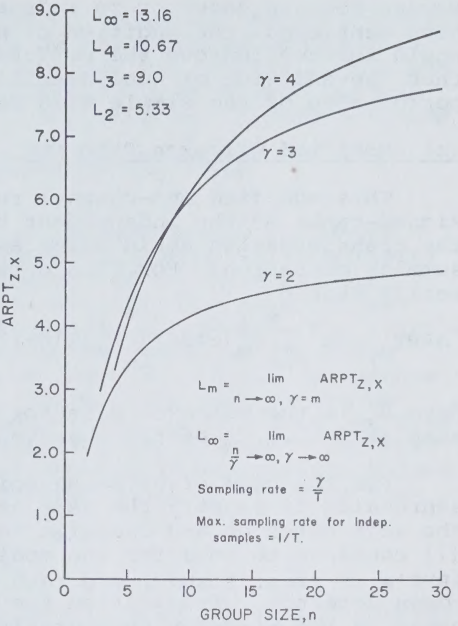


Fig. 2 $\mathrm{ARPT}_{Z,X}$ for sign and Wilcoxon detectors, case (ii)

Table I Relative performance of adaptive threshold sign detector D_Y, case (i)

γ	2	3	4	$\lim_{\gamma\to\infty}$
$\mathrm{ARPT}_{Y,X}$	1.42	1.59	1.68	1.85

Table II Relative performance of adaptive threshold sign detector D_Y, case (ii)

γ	2	3	4	$\lim_{\gamma\to\infty}$
$\mathrm{ARPT}_{Y,X}$	3.36	5.25	6.72	12.83

case (i): Gaussian noise, autocorrelation function

$$\rho_N(\tau) = \begin{cases} (1-|\tau|/T)^2\,[1+|\tau|/(2T)] & ,|\tau| \le T \\ 0 & ,|\tau| > T \end{cases}$$

case (ii): Gaussian noise, autocorrelation function

$$\rho_N(\tau) = \begin{cases} (1-|\tau|/T)^2\,[1+|\tau|/(2T)]\cos(2\pi\tau/T) & ,|\tau| \le T \\ 0 & ,|\tau| > T \end{cases}$$

Neyman-Pearson detector to independent sampling only. As we have mentioned, the addition of pre-detection nonlinearities could further improve the performance of D_Z. It should be noted that the addition of nonlinearities cannot similarly improve the performance of the simple sign detector D_X.

(b) Modified Wilcoxon Detector

This modified one-channel rank detector D_Z is based on the signed-ranks of the independent transformed univariate values, the transformation again being assumed to be effected through a summing operation. For this detection scheme it can be shown easily that

$$\text{ARPT}_{Z,X}=n^2\left[\int_{-\infty}^{\infty} f_Z^2(z)\,dz\Big/\int_{-\infty}^{\infty} f_N^2(x)\,dx\right]^2[\gamma/(n+\gamma-1)] \ . \qquad (9)$$

Here D_X is the Wilcoxon detector operating on the independent samples $X_1,\ldots,X_N$ of the observed waveform.

For the case of Gaussian noise it can be shown that this expression is exactly the same as that obtained in Part (a) for the sign detector and Gaussian noise. This means that Eqs. (5)-(7) continue to hold for the modified Wilcoxon detector D_Z. The ARPT's of Figs. 1 and 2 are thus valid for the case of the Wilcoxon detector. In addition the adaptive threshold detector D_Y based on the Wilcoxon test statistic does not perform as well as in the case of the sign detector, so that the relative performance of D_Z here is even more attractive.

For non-Gaussian processes and large n we can once again approximate $f_Z(z)$ by the Gaussian probability density function, and we get

$$\text{ARPT}_{Z,X}\approx[n^2/(4\pi\sigma_N^2)]\left[n+2\sum_{i=1}^{\gamma-1}(n-i)\rho_N(iT/\gamma)\right]^{-1}\left[\int_{-\infty}^{\infty} f_N^2(x)\,dx\right]^{-2}$$

(large n)

$$\cdot\ [\gamma/(n+\gamma-1)] . \qquad (10)$$

For the case of a noise process with a uniform univariate probability density function, we have from the above that $\text{ARPT}_{Z,X}\approx(3/\pi)$[$\text{ARPT}_{Z,X}$ for the corresponding Gaussian case] and for the double-exponential noise probability density function we have $\text{ARPT}_{Z,X}\approx(2/\pi)$[$\text{ARPT}_{Z,X}$ for the corresponding Gaussian case].

Here again it can be shown that these approximations are very good for moderate values of n, especially if the correlation coefficient between adjacent samples is not too large.

One important advantage of the modified Wilcoxon detector D_Z is that it requires a smaller number of data values to be ranked. For example, suppose N=100 and we are using $\gamma=2$ and n=3. Then $N\gamma/(n+\gamma-1)=50$. In this case we have to rank only 50 numbers instead of 100, a considerable saving.

It should be noted that, instead of basing the test statistic on the sums of the components of the multivariate samples $\underline{Z}_i$, other transformations could be used; for example, one which results in equal ranks being attached to hyperspheres instead of the hyperplanes we have used. The analysis of such a scheme is, however, more involved.

IV. DEPARTURES FROM IDEAL ASSUMPTIONS

(a) Non-Skipped Case

The skipping of $(\gamma-1)$ samples between groups of length n was introduced in the construction of D_Z since, for simplicity of analysis, the multivariate samples had to be independent. In most cases it would not be necessary to skip samples, because, if large enough groups of samples are summed, the correlations between adjacent group-sums become small enough to be negligible. This is sufficient to imply negligible dependence for a large class of processes. In Part (b) we will see an important example of this. In many cases, n also does not have to be very large before this assumption can be made. The ARPT values we have derived would then represent conservative values for the relative performances, because there would be no loss due to skipping.

(b) $\rho_N(\tau)$ Not of Finite Support

As a specific example, consider the stationary Gauss-Markov process, where $\rho_N(\tau)=\exp(-a|\tau|)$. It is well-known that for sampling at the rate a/π the samples can be considered to be independent for nonparametric detection applications [4,9]. With a/π as the base sampling rate it can be shown that the correlation coefficient c between consecutive summed samples Z^i and Z^{i+1} with $\gamma-1$ skipped samples between groups is

$$c = e^{-\pi}(1-e^{-n\pi/\gamma})^2/[n(1-e^{-2\pi/\gamma})+2e^{-\pi/\gamma}(e^{-n\pi/\gamma}-1)] \quad . \qquad (11)$$

For $n=2$ and $\gamma=2$, we find that $c=0.028$, which is negligible.

If samples are not skipped the value of c would be larger. Even so, for $n=5$ and $\gamma=2$, we have $c \approx 0.05$. This is again small enough to be negligible.

We conclude that the assumptions introduced in the beginning of this paper about $\rho_N(\tau)$ do not have to be strictly true for the performance values obtained to be valid. In fact, as was mentioned above, the computed ARPT values may actually be lower than can be obtained with D_Z in practice. For the example above we find that, on using Eq. (4) appropriately modified (i.e. with no skipping loss) for $n=5$ and $\gamma=2$, the value for $\text{ARPT}_{Z,X}$ is 1.45.

It has been assumed in this section that the test statistics always remain asymptotically normally distributed. For the case of the modified sign detector this assumption can be established for a wide class of strong mixing input processes. The case of the modified Wilcoxon detector is more difficult, with available proofs of the asymptotic normality of the test statistic applicable only to the case of strong mixing Gaussian noise processes.

V. CONCLUSION

A class of detectors which retain nonparametric operation under conditions of dependent input data has been described. Particular detectors in this class have been shown to be effective in obtaining higher detection efficiencies than can be obtained by standard nonparametric detectors, since they need not be restricted to operate on independent samples of continuous-time inputs. Specific examples have shown that in many cases considerable improvements in performance can result with sampling rates which are two to four times the rate required for independent samples. Even higher efficiencies could be obtained by detectors which are based on nonlinear transformations rather than the linear transformations that were considered here. The detectors considered here retain to a large degree the simple implementations of standard nonparametric detectors, which make them attractive as practical alternatives.

ACKNOWLEDGEMENTS

This research is supported by the National Science Foundation under Grant GK-24187 and by the U.S. Army Research Office under Contract DAHC04-69-0012.

REFERENCES

1. S.S. Wolff, J.L. Gastwirth and H. Rubin, "The effect of autoregressive dependence on a nonparametric test," IEEE Trans. on Information Theory, Vol. IT-13, pp. 311-313, April 1967.

2. L.D. Davisson, E.A. Feustel and J.W. Modestino, "The effects of dependence on nonparametric detection," IEEE Trans. on Information Theory, Vol. IT-16, pp. 32-41, January 1970.

3. M. Kanefsky and J.B. Thomas, "On adaptive nonparametric detection systems using dependent samples," IEEE Trans. on Information Theory, Vol. IT-11, pp. 521-526, October 1965.

4. J.W. Modestino, "Nonparametric and adaptive detection of dependent data," Ph.D. dissertation, Dept. Elec. Eng., Princeton University, Princeton, N.J., 1969.

5. V.A. Simakhin, "Nonparametric signal detectors when the signal samples are dependent," Telecommunication and Radio Engineering, Vol. 28, pp. 66-72, January 1973.

6. M. Kanefsky, "An automatic decision threshold for polarity coincidence arrays," Proc. Nat'l Electronics Conf., Vol. 22, pp. 2-6, 1966.

7. S.S. Wolff and M.B. Sirvanci, "'Studentizing' rank tests for dependent data," Proc. Seventh Princeton Conf. on Information Sciences and System, pp. 63-66, 1973.

8. M.N. Woinsky and L. Kurz, "Nonparametric detection using dependent samples," IEEE Trans. on Information Theory, Vol. IT-16, pp. 355-358, May 1970.

9. M. Kanefsky and J. B. Thomas, "Polarity coincidence correlation using dependent samples," Proc. Nat'l Electronics Conf., Vol. 20, pp. 714-719, 1964.

10. G.E. Noether, "On a theorem of Pitman," Ann. Math. Stat., Vol. 26, pp. 64-68, 1955.

THE OPTIMIZATION AND PERFORMANCE OF DETECTORS BASED ON PARTITION TESTS

W. J. BUSHNELL AND L. KURZ*
BELL LABORATORIES
Whippany, New Jersey 07981

ABSTRACT

The Partition Test Statistic for IID samples in specified noise is shown to be simple to implement and yet robust. The equivalence of the Partition Test and the Likelihood Ratio Test as the number of partitions increases is shown. An optimal partitioning algorithm is discussed and the probability of error expression and bounds are formulated and evaluated.

INTRODUCTION AND SUMMARY

A Partition Test is a hypothesis testing procedure which is a generalization of the classic sign test. The use of multiple partitions was first investigated in 1957 by Bloom et al. [1]. However, the completely general m-interval Partition Test was proposed by Ching [2] in 1972. Previous research in Partition Test [2,3,4] has been concerned with its nonparametric aspects. This paper summarizes the results of recent investigations into the optimization and performance of Partition Tests in specified noise environments.

The Partition Test Statistic is formulated using a single partitioning of the sample space for the binary hypothesis testing problem. The test statistic is shown to have a combination of properties (i.e., robustness and ease of implementation) which make it useful in many practical problems. The equivalence of the Partition Test and the Likelihood Ratio Test as the number of partitions goes to infinity is shown. An expression for the error probability is derived, and a tight, easily computed, upper bound is presented. The selection of the partitions which yield minimum error probability is discussed. A theorem on the symmetry of the partitions for a symmetrical noise distribution and a shift alternative is proved. The optimal partitions for the shift alternative in Gaussian noise are computed and the probability of error is calculated. The error rates are shown to rapidly converge from that of the Sign Test to that of the Likelihood Ratio Test as the number of partitions is increased.

THE PARTITION TEST STATISTIC

Assume $X = (x_1, x_2, \ldots, x_L)$ is a set of IID data samples which are to be used to test whether the samples were generated under hypothesis H_0 with CDF $F_x(x|H_0)$ or whether the samples were generated under the alternative H_1 with CDF $F_x(x|H_1)$. The distribution functions are assumed to be continuous and strictly increasing. $A = (a_0, a_1, \ldots, a_m)$ is a vector of ordered constants

---

[*] Professor of Electrical Engineering at the Polytechnic Institute of New York.

which partitions the sample space of the random variables into a set of m-disjoint intervals. It is assumed that the partitions are judiciously chosen so that the probability of a data sample lying in any interval is non-zero under H_0 and H_1 and so that the union of the intervals contains the sample space under H_0 and H_1. Define the m-tuple $N = (n_1, n_2, \ldots, n_m)$ as the number of data samples in each interval.

The test statistic, $T(N)$, is based upon the observation of the m-tuple, N, and is formulated in terms of the logarithm of the likelihold ratio.

$$T(N) = \ln[P\{N|H_1\}/P\{N|H_0\} \qquad (3.1)$$

The random variables $n_1, \ldots, n_m$ are multinominally distributed [5]

$$P\{N|H_i\} = L! \prod_{j=1}^{m} \frac{(p_{ij})^{n_j}}{n_j!} \qquad (3.2)$$

where p_{ij} is the probability of a data sample lying in the $j^{\underline{th}}$ interval given hypcthesis H_i.

$$p_{ij} = F_x(a_j|H_i) - F_x(a_{j-1}|H_i) \qquad (3.3)$$

Since $p_{ij} \neq 0$ for all i and j the logarithm of the likelihood ratio can be written as

$$T(N) = N \cdot B = \sum_{j=1}^{m} n_j b_j \qquad (3.4)$$

where $B = (b_1, \ldots, b_m)$ is the score vector given by

$$b_j = \ln(p_{1j}/p_{0j}) \qquad (3.5)$$

The following theorem previews the high efficiency of the Partition Test which is shown in Section 5. It proves that the error rate can be reduced to the theoretical minimum by increasing the number of partitions.

THEOREM 1: Given the following conditions on a binary hypothesis testing problem

1) Under both the hypothesis and the alternative the observation $(x_1, \ldots, x_L)$ are IIDRV from a sample space Ω.

2) The cumulative distribution functions under the hypothesis, $F_X(x|H_0)$, and under the alternative, $F_X(x|H_1)$, are continuous and strictly increasing.

3) The partitions $A = (a_0, \ldots, a_m)$ are the m + 1 quantiles of the joint distribution

$$A = \left\{ a_j = H_X^{-1}(j/m): \quad j = 0,1,\ldots,m \right\}$$

where $H_X(x) = \alpha\, F_X(x|H_0) + (1-\alpha)F_X(x|H_1)$ for any α such that $0 < \alpha < 1$.

Then as $m \to \infty$ the partition test statistic, $T(N)$, becomes equivalent to the logarithm of the Likelihood Ratio Statistic, $\Lambda(X)$,

$$\lim_{m\to\infty} T(N) = \ln[\Lambda(X)]$$

PROOF: The Partition Test Statistic, $T(N)$, is given by

$$T(N) = \sum_{j=1}^{m} n_j \ln\left(\frac{p_{1j}}{p_{0j}}\right) \tag{3.6}$$

Instead of considering each interval, consider each random variable

$$T(N) = \sum_{k=1}^{L} \ln\left(\frac{p_{1j}(k)}{p_{0j}(k)}\right) \tag{3.7}$$

where

$$p_{ij}(k) = P\{a_{j-1} < x_k \leq a_j | H_i\} \tag{3.8}$$

since $p_{ij}(k) > 0$ and ln is a continuous function

$$\lim_{m\to\infty} T(N) = \sum_{k=1}^{L} \ln\left[\lim_{m\to\infty} \frac{p_{1j}(k)}{p_{0j}(k)}\right] \tag{3.9}$$

Since the partitions are the m+1 quantiles and $F_X(x|H_0)$ and $F_X(x|H_1)$ are continuous and strictly increasing, as m increases the interval containing x_k decreases. By L'Hospital's Rule Equation (3.9) becomes

$$\lim_{m\to\infty} T(N) = \sum_{k=1}^{L} \left[\ln \frac{P\{X = x_k|H_1\}}{P\{X = x_k|H_0\}}\right] = \ln[\Lambda(x)] \tag{3.10}$$

The decision rule for the Partition Test Statistic is

$$\text{IF} \quad N\cdot B > \ln(\Lambda_0) \quad \text{select } H_1$$

$$\text{IF} \quad N\cdot B < \ln(\Lambda_0) \quad \text{select } H_0$$

where Λ_0 is chosen in some appropriate fashion (i.e., Bayes, Neyman-Pearson, or Minimax Criterion). Ties are resolved by random selection. The probability of error is

$$P_e = \sum_{\{N_1\}} P\{N|H_0\}P\{H_0\} + \sum_{\{N_2\}} P\{N|H_1\}P\{H_1\} + \frac{1}{2} \sum_{\{N_3\}} P\{N\} \tag{3.11}$$

where $\{N_1\}$, $\{N_2\}$ and $\{N_3\}$ are sets of m-tuples which satisfy the following:

$$\{N_1\} = \{N : N \cdot B > \ln(\Lambda_0)\}$$

$$\{N_2\} = \{N : N \cdot B < \ln(\Lambda_0)\}$$

$$\{N_3\} = \{N : N \cdot B = \ln(\Lambda_0)\} .$$

$P\{H_0\}$ and $P\{H_1\}$ are the apriori hypothesis probabilities.

For a small number of data samples and a small number of intervals, Equation (3.11) can be methodically evaluated by considering each m-tuple and summing its probability of occurrence under the correct hypothesis. However, if the number of possible m-tuples is extremely large, it is more convenient to use the upper bound on error probability which is presented in the following theorem.

THEOREM 2: Given the Partition Test Statistic T(N) and that $E\{T|H_0\} \leq \ln(\Lambda_0)$ and $E\{T|H_1\} \geq \ln(\Lambda_0)$ then the probability of error of the test is bounded by

$$P_e \leq P\{H_0\} \inf_{\omega} \left[\Lambda_0^{-\omega} \left(\sum_{i=1}^{m} \left(\frac{p_{1i}}{p_{0i}} \right)^{\omega} p_{0i} \right) \right]^L + P\{H_1\} \inf_{\omega} \left[\Lambda_0^{-\omega} \left(\sum_{i=1}^{m} \left(\frac{p_{1i}}{p_{0i}} \right)^{\omega} p_{1i} \right) \right]^L \qquad (3.12)$$

where the infinum is with respect to all real values of ω.

PROOF: If $E\{T|H_0\} \leq \ln(\Lambda_0)$ and $E\{T|H_1\} \geq \ln(\Lambda_0)$ then by applying the Chernoff Bound [6] Equation (3.11) can be bounded as

$$P_e \leq P\{H_0\} g_T(\ln(\Lambda_0)|H_0) + P\{H_1\} g_T(\ln(\Lambda_0)|H_1) \qquad (3.13)$$

where

$$g_T(\ln(\Lambda_0)|H_j) = \inf_{\omega} [\exp(-\ln(\Lambda_0)\omega) G_T(\omega|H_j)] . \qquad (3.14)$$

$G_T(\omega|H_j)$ is the moment generating function of the random variable T given hypothesis H_j. The moment generating function for T is given by Cramer [5].

$$G_T(\omega|H_j) = \left[\sum_{i=1}^{m} \exp(b_i \omega) p_{ji} \right]^L \qquad (3.15)$$

OPTIMAL PARTITIONS

The partial derivative of the error probability with respect to the partitions yields a necessary condition on the optimal partitions,

$$\begin{aligned}\frac{\partial P_e}{\partial a_j} &= P\{H_0\} \sum_{\{N_1\}} P\{N|H_0\}\left[\frac{n_j}{p_{0j}} - \frac{n_{j+1}}{p_{0,j+1}}\right] f_x(a_j|H_0) \\ &+ P\{H_1\} \sum_{\{N_2\}} P\{N|H_1\}\left[\frac{n_j}{p_{1j}} - \frac{n_{j+1}}{p_{1,j+1}}\right] f_x(a_j|H_1) \\ &+ \frac{1}{2} \sum_{\{N_3\}} P\{N|H_1\}\left[\frac{n_j}{p_{1j}} - \frac{n_{j+1}}{p_{1,j+1}}\right] f_x(a_j|H_1) \\ &= 0 \qquad (4.1)\end{aligned}$$

where $f_x(\cdot|H_i)$ is the probability density function under H_i. Because of the complicated functional relationships which are likely to arise because of $f_x(\cdot|H_0)$ and $f_x(\cdot|H_1)$ it is not likely to find a closed form solution for the optimal partitions. Hence it is necessary to use a gradient search technique. The following theorem proves, however, that for the shift alternative in a symmetrical noise distribution the partitions are symmetrical. Thus under these conditions half of the partitions can be determined from their symmetrical counterparts.

THEOREM 3: Given the following conditions

1) The observations $(x_1,\dots,x_L)$ are IIDRV from a sample space Ω.

2) The hypothesis and alternative are equally likely and have cumulative distribution functions $F_X(x|H_0)$ and $F_X(x|H_1)$ respectively.

3) $F_X(x|H_0)$ and $F_X(x|H_1)$ are continuous and symmetrical in the sence that

$$F_X(x|H_0) = 1 - F_X(\mu_1 + \mu_0 - x|H_1)$$

where μ_i is the mean under H_i.

4) $A = (a_0,\dots,a_m)$ are symmetric partitions in the sense that

$$a_{m-j} = \mu_1 + \mu_0 - a_j \ .$$

5) The j^{th} component of the partition A is optimal in that

$$\left.\frac{\partial P_e}{\partial a_j}\right|_A = 0$$

Then the m-j$^{\underline{th}}$ component is also optimal

$$\left.\frac{\partial P_e}{\partial a_{m-j}}\right|_A = 0 \; .$$

PROOF: The partial derivative of the probability of error with respect to the i$^{\underline{th}}$ partition is given by Equation (4.1). Given that the distributions and the partitions are symmetric the following three properties can be shown.

Property 1

$$f_X(a_j|H_0) = f_X(a_{m-j}|H_1)$$

Property 2

$$p_{0j} = p_{1,m-j+1}$$

Property 3

$$\text{a)} \qquad \sum_{\{N_1\}} P\{N|H_0\}n_j = \sum_{\{N_2\}} P\{N|H_1\}n_{m-j+1}$$

$$\text{b)} \qquad \sum_{\{N_3\}} P\{N|H_0\}n_j = \sum_{\{N_3\}} P\{N|H_0\}n_{m-j+1}$$

Given the j$^{\underline{th}}$ component is optimal then by Properties 1, 2, and 3, Equation (4.1) can be written as

$$\begin{aligned}
&\sum_{\{N_2\}} P\{N|H_1\}\left[\frac{n_{m-j}}{p_{1,m-j}} - \frac{n_{m-j+1}}{p_{1,m-j+1}}\right] f_x(a_{m-j}|H_1)\Bigg|_A \\
&+ \sum_{\{N_1\}} P\{N|H_0\}\left[\frac{n_{m-j}}{p_{0,m-j}} - \frac{n_{m-j+1}}{p_{0,m-j+1}}\right] f_x(a_{m-j}|H_0)\Bigg|_A \\
&+ \sum_{\{N_3\}} P\{N|H_1\}\left[\frac{n_{m-j}}{p_{1,m-j}} - \frac{n_{m-j+1}}{p_{1,m-j+1}}\right] f_x(a_{m-j}|H_1)\Bigg|_A \\
&= 0 \qquad\qquad (4.2)
\end{aligned}$$

Hence

$$\left.\frac{\partial P_e}{\partial a_{m-j}}\right|_A = 0 \; .$$

NUMERICAL RESULTS

The probability of error expression (3.11) is evaluated for the shift alternative in Gaussian noise as a function of the signal-to-noise power ratio for L = 2 samples and L = 5 samples, Figures 1 and 2 respectively. The probability of error of the Likelihood Ratio Test is shown to lower bound the error performance. The partitions are selected by the gradient search technique so as to minimize the probability of error. It is assumed that $P\{H_0\} = P\{H_0\} = \frac{1}{2}$ and that $\Lambda_0 = 1$. Note, in Figures 1 and 2 that as the number of intervals in the partition test increases the performance of the test rapidly approaches the performance of the Likelihood Ratio Test.

The efficiency of the Chernoff Bound is demonstrated in Figures 3 and 4. The exact probability of error expression and the Chernoff Bound are evaluated as a function of the number of samples for S/N = 1.0 and S/N = 3.0, for m = 2 intervals and m = 4 intervals. In all cases the partitions were selected, by means of a gradient search technique, to minimize the exact probability of error expression. These partitions were then used in evaluating the Chernoff Bound.

REFERENCES

1. Bloom, F. J., Chang, S. S. L., Harris, B., Hauptschein, A., Morgan, K. C., "Improvement of Binary Transmission by Null-Zone Reception," Proc. IRE, July 1957, pp. 963-975.

2. Ching, Y. C. and Kurz, L., "Nonparametric Detectors Based on m-Interval Partitioning," IEEE Trans. on I.T., March 1972, pp. 251-257.

3. Gotz, B., "Quantile Estimation, m-Interval Statistics and Associated Procedures," Ph.D. Thesis, New York University, N. Y., September 1971.

4. Cochrane, J. I., "Some Robust Procedures for Classification and Signal Selection in Unspecified Noise Environments," Ph.D. Thesis, New York University, May 1972.

5. Cramer, H., Mathematical Methods of Statistics, Princeton University Press, 1946.

6. Chernoff, H., "A Measure of Asymptotic Efficiency for Tests of A Hypothesis Based on the Sum of Observations," Annals of Math. Stat., 1952, pp. 493-507.

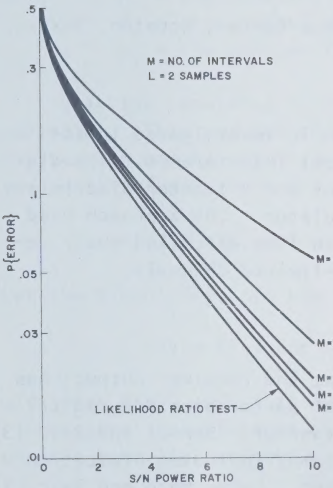


Figure 1

P{error} vs. SNR L = 2 Samples

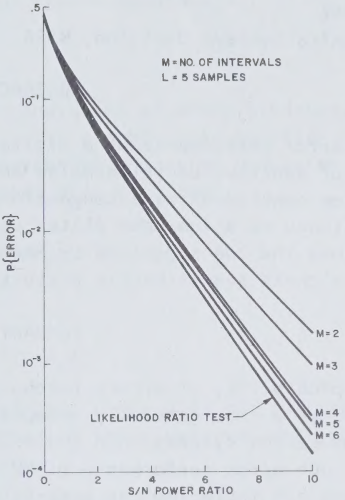


Figure 2

P{error} vs. SNR L = 5 Samples

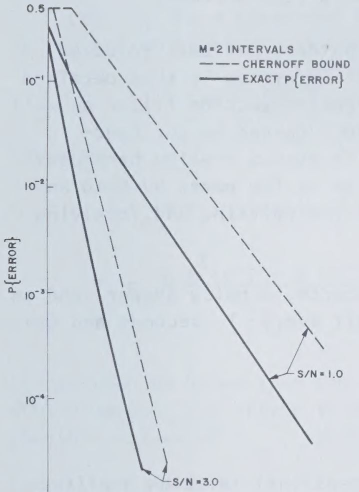


Figure 3

P{error} Bound m = 2 Intervals

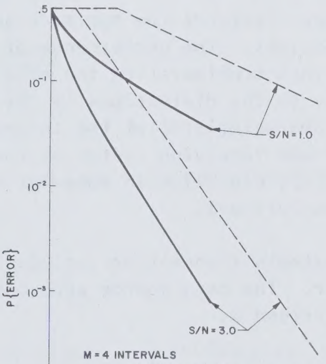


Figure 4

P{error} Bound m = 4 Intervals

THE EFFECT OF INTERSYMBOL INTERFERENCE ON THE PERFORMANCE OF FM DIGITAL SYSTEMS

P. PAPANTONI-KAZAKOS
Dept. of Electrical Engineering, Rice University, Houston, Texas
I. PAZ
Avionics Systems Division, NASA, Johnson Space Center, Houston, Texas

ABSTRACT

The error performance of a digital FM system is investigated in the presence of additive Gaussian noise and intersymbol interference. The digital system considered is a conventional modulator and a limiter discriminator followed by a low-pass filter as the demodulator. The approach used here follows the one suggested by Mazo and Salz in 1966 extending their results to signals significantly distorted by band-limited channels.

SUMMARY

The probability of errors (wrong decisions at the receiver output) has lately become a more generally accepted measure of performance for digital data transmission systems than their S/N ratio transfer. Bennet and Salz [3] analyzed such error performance of FM binary systems, including distortion effects but not including the post-detection filter. Later Mazo and Salz [2] predicted the error performance of digital FM systems with the post-detection filter included using the notion of "clicks" as first introduced by Rice [1] in 1953. No consideration was given, however, to the distortions introduced into the FM waveform by the band-limiting effect of the channel.

This paper extends the approach suggested by Mazo and Salz to cover those distortions. The performance of a digital FM system is thus obtained that takes into consideration the effects of a post-detection filter as well as those due to the distortions in the FM waveform caused by the band-limitation characteristic of the channel. The FM system studied here uses almost the same receiving system as the one given in the paper by Mazo and Salz (Ref. [2], Fig. 1) with somewhat different transmitting and receiving filters (see further).

The system's transmitter includes a data source, a pulse shaper, and an FM modulator. The data source produces one digit every T seconds and can best be described by

$$s(t) = \omega_d \sum_{n=0}^{\infty} a_n g(t-nT) \qquad (1)$$

with ω_d = a constant (with frequency dimensions) relating amplitude to frequency

a_n = the value of the digit transmitted at the time nT

$g(x)$ = a rectantular signaling pulse of unit amplitude

We shall restrict ourselves in this summary (for brevity and more compact formulation) to the binary case, although the results may easily be extended to M-ary cases. The output of the FM modulator is then given by

$$S(t) = A \cos(\omega_c t + \int_o^t s(\tau)d\tau + \theta) \qquad (2)$$

with A = the constant amplitude of the FM waveform
θ = the initial phase
ω_c = the carrier frequency

Only the receiving filter is here considered of sharp band-pass characteristics. This way the channel noise $n(t)$, considered Gaussian and additive, can now be resolved into two independent Gaussian processes $a(t)$ and $b(t)$ of zero mean and some variance σ_o^2, such that

$$n(t) = a(t) \cos \omega_c t - b(t) \sin \omega_c t \qquad (3)$$

Let the signal input to the limiter be

$$\int_1(t) = X(t) \cos \omega_c t - Y(t) \sin \omega_c t \qquad (4)$$

with the $X(t)$ and $Y(t)$ signal coordinates containing all the distortions (intersymbol interference effects) imposed on the signaling waveform by the channel band limitations.

For the binary case, using the above generalized $X(t)$ and $Y(t)$ signal coordinates, the results obtained by Mazo and Salz are changed as follows:

Let λ_n be a parameter that is a function of all the previous history of the system (past digits) with:

$$\lambda_n \triangleq \tan^{-1} \frac{Y(nT)}{X(nT)} - \tan^{-1} \frac{Y(n-1)T)}{X(n-1)T)} \qquad (5)$$

Let also

$$P_o \triangleq P_r \text{ (sample value at } t = nT \text{ is larger than zero)*}$$

$$R_n \triangleq \frac{X^2(nT) + Y^2(nT)}{2\sigma_o^2}$$

It can then be shown that for R_n large (large signal-to-noise ratio) and slowly varying, we obtain three different expressions for three different possible values of λ_n, namely:

i) $\lambda_n < \frac{\pi}{2}$

---

* For which case we decide that a_n is one.

$$P_o = \frac{1}{\sqrt{8\pi}} \frac{\cot\left(\frac{\lambda_n}{2}\right)}{\sqrt{\cos(\lambda_n)}} \frac{\exp\left[-2R_n \sin^2\left(\frac{\lambda_n}{2}\right)\right]}{\sqrt{R_n}} \tag{7}$$

ii) $\lambda_n = \frac{\pi}{2}$ *

$$P_o = 1/4 \exp(-R_n) \tag{8}$$

iii) $\lambda_n > \frac{\pi}{n}$

$$P_o = \frac{\exp\{-R_n[1 + \cos^2(\lambda_n)]\}}{2\pi \sqrt{\pi}\, R_n \sqrt{R_n} \sin(\lambda_n) \cos^2(\lambda_n)} \tag{9}$$

Due to the way we defined λ_n, the error probabilities obtained in (7),(8), (9) will average over the whole past history of the system.

Assuming now we replace the transmission and receiving filters by one single rational equivalent filter, we may represent its impulse response by

$$h(t) = \left[\sum_{k=1}^{M} C_k e^{-b_k t}\right] \cos \omega_c t \tag{10}$$

Assuming a binary situation with $\omega_c \gg \omega_d$, we obtain the following expressions for the coordinates $X(t)$ and $Y(t)$ at time $t = (m+1)T$.

$$\begin{aligned} X(t) = {} & \frac{A}{2} \sum_{k=1}^{M} C'_k \omega_d \sum_{n=1}^{m} \left[e^{-b_k(t-nT)} (a_n - a_{n+1}) \cdot \sin(\theta + \varphi_{dn})\right] \\ & + \frac{A}{2} \sum_{k=1}^{M} b_k (C'_k + d'_k) \cos(\theta + \varphi_{c,m+1}) \\ & + \frac{A}{2} \sum_{k=1}^{M} [C'_k a_{m+1} \omega_d + 2\omega_c d'_k] \sin(\theta + \varphi_{c,m+1}) \end{aligned} \tag{11}$$

$$\begin{aligned} Y(t) = {} & - \frac{A}{2} \sum_{k=1}^{M} C'_k \omega_d \sum_{n=1}^{m} \Big[e^{-b_k(t-nT)} (a_n - a_{n+1}) \cos(\theta + \varphi_{dn}) \\ & - b_k (C'_k + d'_k) \sin(\theta + \varphi_{d,m+1}) \\ & + (C'_k \omega_d a_{m+1} + 2\omega_c d'_k) \cos(\theta + \varphi_{d,m+1})\Big] \end{aligned} \tag{12}$$

with

---

* The only case in which the error from mixing up the signals is comparable to the one obtained for "clicks."

$$
\begin{aligned}
C'_k &= \frac{C_k}{b_k^2 + \omega_d^2} \\
d'_k &= \frac{C_k}{b_k^2 + 4\omega_c^2} \\
\varphi_{dn} &= \omega_d T \sum_{i=0}^{n} a_i \\
\varphi_c &= \omega_c T \sum_{i=0}^{n} a_i
\end{aligned}
\tag{13}
$$

CONCLUSIONS

a. Expressions (7), (8), (9) in conjunction with equations (11) and (12) provide a closed form formulation for the error probability of a binary FSK system with arbitrary transmission filters. The search for an optimal such filter could, therefore, be attempted.

b. Moreover, those formulas hold also for situations in which the distortions introduced into the waveform by intersymbol interference cannot be neglected.

c. Equations (8) and (9) remain the same and the expressions for error probabilities are easily extendable to the multilevel FSK systems.

d. It can easily be shown that λ_n is independent of the initial phase θ .

e. Significant simplifications in equations (8) and (9) are possible for large values of b_k .

REFERENCES

1. Rice, S.O., "Noise in FM receivers," Time Series Analysis, Chapter 25, M. Rosenblatt, ed., John Wiley and Sons, Inc., New York, 1963.

2. Mazo, J.E., and J. Salz, "Theory of error rates for digital FM," B.S.T.J. XLV: 1511-1535, 1966.

3. Bennett, W.R., and J. Salz, "Binary data transmission by FM over a real channel," B.S.T.J. 42: 2387-2426, 1963.

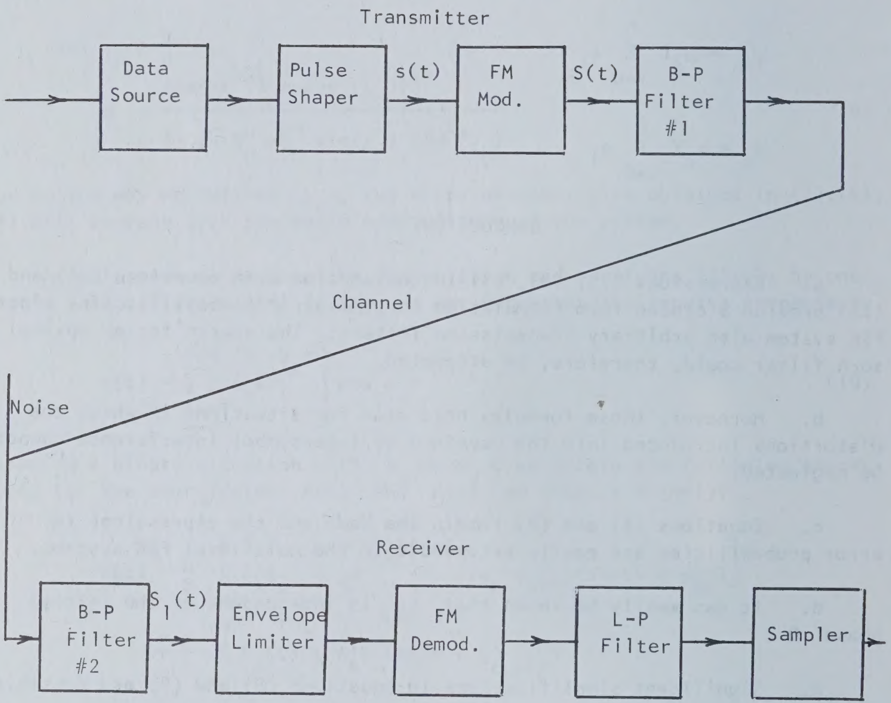

Transmitter
Data Source
Pulse Shaper
s(t)
FM Mod.
S(t)
B-P Filter #1
Channel
Noise
Receiver
B-P Filter #2
$S_1(t)$
Envelope Limiter
FM Demod.
L-P Filter
Sampler

FIG. 1

SOME RESULTS ON THE CHARACTERIZATION OF RANDOM SEQUENCES AND RELATED PULSE TRAINS

G.L. WISE AND J.B. THOMAS
Department of Electrical Engineering
Princeton University, Princeton, N.J. 08540

ABSTRACT

A class of stationary Markov sequences is considered. The diagonal series expansion of the bivariate density corresponding to any two random variables in the Markov sequence is derived. Some properties of pulse trains are also considered, where the amplitudes of the pulses form a Markov sequence.

I. INTRODUCTION

Often, as a first step in the analysis of a problem involving a sequence of random variables , the random variables are assumed to be independent and identically distributed. With a more exacting analysis they may be modeled differently; however, the model is usually limited by limitations in the theory of multivariate statistics. Often only a second moment description of the sequence is assumed. This restricts one to second moment operations which are often constrained to be linear. At other times, the assumption is made that the sequence forms a discrete time Gaussian process. This assumption is sometimes made more for the tractability of Gaussian processes than for the appropriateness of the model. Frequently the sequence is assumed to be a stationary Markov sequence. This assumption is often intuitively appealing from a physical viewpoint. By the Markov property a complete statistical description of the sequence can be obtained from the bivariate density of two adjacent random variables.

In this paper a class of stationary Markov sequences is considered. The diagonal series expansion of the bivariate density corresponding to any two random variables in the Markov sequence is derived, and some second moment properties of the sequence are examined. Some properties of pulse trains are also considered, where the amplitudes of the pulses form a Markov sequence.

II. A CHARACTERIZATION OF THE BIVARIATE DENSITY

Let X_n be a second-order stationary, discrete-time random process, and let $h(x,y)$ denote the bivariate density of $X=X_n$ and $Y=X_{n+1}$. Assume that $h(x,y)=h(y,x)$, and let $f(x)$ denote the univariate density. We will assume that $h(x,y)$ satisfies the following weak integrability condition:

$$\iint \frac{[h(x,y)]^2}{f(x)f(y)}\, dxdy < \infty.$$

With this assumption, $h(x,y)$ can be written in the diagonal series expansion [1,2]

$$h(x,y) = f(x)f(y) \sum_{n=0}^{\infty} a_n \theta_n(x)\theta_n(y) \quad . \tag{1}$$

In this expansion, $\{\theta_n(x)\}$ is a sequence of functions that are orthonormal with respect to the weight function $f(x)$; $\theta_0(x)\equiv 1$; and the coefficient sequence is given by

$$a_n = E\{\theta_n(X)\theta_n(Y)\} \; .$$

The convergence is in the sense that

$$\lim_{N\to\infty} \iint \left[\frac{h(x,y)}{f(x)f(y)} - \sum_{n=0}^{N} a_n \theta_n(x)\theta_n(y)\right]^2 f(x)f(y)\,dxdy = 0 \; .$$

Barrett and Lampard [3] first noticed the engineering significance of diagonal series expansions of bivariate density functions. They considered the class Λ, which is the class of bivariate density functions such that, in the diagonal series expansion, the n-th orthonormal function is an n-th degree polynomial. The orthonormal polynomials were obtained by using the Gram-Schmidt procedure to orthonormalize the sequence $\{1,x,x^2,x^3,\ldots\}$. For a class Λ bivariate density, the correlation coefficient is given by a_1. Also, the minimum mean squared error estimate of one random variable in terms of the other will always be linear. As an example, consider the symmetric bivariate Gaussian density, with mean μ, variance σ^2, and correlation coefficient ρ. In this case [3], $a_n = \rho^n$ and

$$\theta_n(x) = \frac{H_n\left(\frac{x-\mu}{\sigma}\right)}{\sqrt{n!}} \; ,$$

where $H_n(x)$ is the n-th Hermite polynomial.

In general, the orthonormal functions will not be polynomials. Consider the following expansion [4]:

$$\sum_{n=0}^{\infty} (-1)^n \epsilon_n J_{2n}(\lambda)\cos(2nx)\cos(2ny) = \cos(\lambda \cos x \cos y)\cos(\lambda \sin x \sin y) \, ,$$

where

$$\epsilon_n = \begin{cases} 1 \; , & n=0 \\ 2 \; , & n\geq 1 \; . \end{cases}$$

With a slight rearrangement, we can obtain the diagonal series expansion of the bivariate density

$$h(x,y) = \frac{\cos(\lambda \cos x \cos y)\cos(\lambda \sin x \sin y)}{4\pi^2 \, J_0(\lambda)} \; ,$$

for $x,y \in [-\pi,\pi]$, and where λ is a parameter belonging to $[0,\pi/2]$. In this case we have

$$f(x) = \frac{1}{2\pi} \quad ,$$

for $x \in [-\pi,\pi]$,

$$a_n = (-1)^n \frac{J_{2n}(\lambda)}{J_0(\lambda)} \quad ,$$

and

$$\theta_n(x) = \sqrt{\epsilon_n} \cos 2nx \ .$$

For notational convenience, define the kernel

$$A(x,y) = \frac{h(x,y)}{\sqrt{f(x)f(y)}} \quad ,$$

where $A(x,y) = 0$ if either $f(x)$ or $f(y)$ is zero. Let A denote the operator induced by the kernel $A(x,y)$; that is, $v=Au$ is defined on the space of measurable square integrable functions by

$$v(x) = \int A(x,y)u(y)dy.$$

It can be shown [5] that A is a self-adjoint completely continuous operator.

Let $\phi_n(x) = \sqrt{f(x)}\ \theta_n(x)$. It follows that

$$A\phi_n = a_n\phi_n \ .$$

Thus $\{\phi_n\}$ and $\{a_n\}$ are the (orthonormal) eigenfunctions and eigenvalues of the operator A. Notice that

$$A(A\phi_n) = A(a_n\phi_n) = a_n^2\phi_n \ .$$

In general, for any positive integer k, we have

$$A^k\phi_n = a_n^k\phi_n$$

where the operator A^k and the kernel $A^k(x,y)$ are defined by

$$\begin{aligned} A^k u &= \int A^k(x,y)u(y)dy \\ &= A(A^{k-1}u) \\ &= \iint A(x,y)A^{k-1}(y,z)u(z)dzdy, \end{aligned}$$

and where $A^1 = A$. The kernel $A^k(x,y)$ is simply the k-th iterate of the kernel $A(x,y)$. It follows [6] that

$$A^k(x,y) = \sum_{n=0}^{\infty} a_n^k \phi_n(x)\phi_n(y) ,$$

where the convergence is in the sense that

$$\lim_{N\to\infty} \iint [A^k(x,y) - \sum_{n=0}^{N} a_n^k \phi_n(x)\phi_n(y)]^2 \, dxdy=0.$$

Let a function $h(x,y,k)$ be defined by

$$h(x,y,k) = \sqrt{f(x)f(y)} \; A^k(x,y).$$

Then we have

$$h(x,y,k) = \int \frac{h(x,z)h(y,z,k-1)}{f(z)} dz, \tag{2}$$

where $h(x,y,1) = h(x,y)$. From this we see that $h(x,y,k)$ is a valid bivariate density function. Also, the diagonal series expansion is given by

$$h(x,y,k) = f(x)f(y) \sum_{n=0}^{\infty} a_n^k \theta_n(x)\theta_n(y).$$

Therefore, we conclude that, in the diagonal series expansion of a symmetric bivariate density, the coefficient sequence can be raised to any positive integral power and the resulting diagonal series expansion will also be that of a valid bivariate density function.

Rewriting Eq. (2) as

$$\frac{h(x,y,k)}{f(x)} = \int \frac{h(x,z)}{f(x)} \frac{h(z,y,k-1)}{f(z)} dz,$$

we recognize this equation as the Chapman-Kolmogorov equation [7,p. 531]. This yields the following theorem:

Theorem 1: Let X_n be a second-order stationary, discrete-time random process. Assume that the bivariate density of X_n and X_{n+1} is given by Eq. (1). Then X_n is a Markov process if and only if, for any positive integer k, the bivariate density of X_n and X_{n+k} is given by

$$h(x,y,k) = f(x)f(y) \sum_{n=0}^{\infty} a_n^k \theta_n(x)\theta_n(y).$$

Many bivariate densities possess a diagonal series expansion where the coefficients are of the form $a_n=\lambda^n$. For example, if the orthonormal functions $\theta_n(x)$ correspond to the orthonormalized Hermite polynomials, generalized Laguerre polynomials, or Jacobi polynomials, and if the univariate densities are chosen appropriately [8], then a bivariate density in the form of Eq. (1) exists where the coefficients are given by $a_n=\lambda^n$, where λ is any number in $[0,1)$. In the case of orthonormalized Hermite polynomials or Gegenbauer polynomials [8], the parameter λ can be any number in $(-1,1)$. Recall that Legendre polynomials and both kinds of Chebychev polynomials belong to the class of Gegenbauer polynomials. Also, it is not hard to show that if the orthonormal functions $\theta_n(x)$ are uniformly bounded (in both x and n) then a bivariate density will always exist where the coefficients are of the form $a_n=\lambda^n$. Naturally, any strictly monotonic transformation on the random variables will not change the coefficient sequence. The class of Markov sequences constructed from bivariate densities of this type is such that the bivariate density for any two random variables in a sequence will have the same basic functional form as that for two adjacent random variables in the sequence.

Throughout the sequel we will assume that X_n is a Markov process. Thus all statistical properties of the sequence can be derived from the bivariate density. In many cases, Theorem 1 furnishes an explicit and relatively simple method for establishing several properties of the Markov sequence. We will study some properties of the bivariate density h(x,y) and then extend them to the sequence X_n.

III. SOME SECOND MOMENT PROPERTIES

Consider two random variables X and Y having the bivariate density given in Eq. (1). Let g(x) be a Baire function such that

$$\int [g(x)]^2 f(x)dx < \infty.$$

Then it can be shown that

$$E\{g(X)g(Y)\} = \sum_{n=0}^{\infty} b_n^2 a_n ,$$

where

$$b_n = \int g(x)\theta_n(x)f(x)dx. \tag{3}$$

Now consider the discrete-time random process given by $g(X_n)$. It follows from the preceding statements and from Theorem 1 that

$$E\{g(X_{n+k})g(X_n)\} = \sum_{n=0}^{\infty} b_n^2 a_n^k , \tag{4}$$

where k is a positive integer.

We now consider the estimation of g(Y) given X, where X and Y have the bivariate density given in Eq. (1). It can be shown that the minimum mean squared error estimate of g(Y) given X is

$$E\{g(Y)|X\} = \sum_{n=0}^{\infty} a_n b_n \theta_n(X),$$

where b_n is given by Eq. (3) and where the convergence is in the quadratic mean. Now consider the random sequence X_n. From the preceding statement and from Theorem 1, we see that the minimum mean squared error predictor of $g(X_{n+k})$ given $X_n, X_{n-1}, X_{n-2}, \ldots$, is

$$E\{g(X_{n+k})|X_n,X_{n-1},X_{n-2},\ldots\} = \sum_{j=0}^{\infty} a_j^k b_j \theta_j(X_n). \tag{5}$$

In the remainder of this chapter we will assume that X_n is a second-order sequence with variance σ^2. Let μ denote the expectation of X_n and let the correlation coefficient of X_n and X_{n+1} be denoted by ρ. Let the autocovariance function of the sequence X_n be given by R(k).

Theorem 2: Consider the Markov sequence X_n with autocovariance function R(k). Either R(k)=0 for all nonzero integers k, or R(2k)>0 for all integers k.

Proof: Letting $g(x)=x-\mu$, we see that the autocovariance function R(k) is given by Eq. (4) when k is a positive integer. Thus for any nonzero integer k, we have

$$R(2k) = \sum_{n=0}^{\infty} b_n^2 a_n^{2|k|} \geq 0.$$

Therefore R(2k) = 0 if and only if

$$b_n a_n = 0 \quad \text{for} \quad n=0,1,2,\ldots.$$

The result follows. ||

Theorem 2 asserts that either the Markov sequence is white in the sense that it is a sequence of uncorrelated random variables, or else, in a second-moment sense, the sequence never has finite memory.

It is well known [9] that if X_n is a stationary Gaussian Markov sequence then the autocovariance function is given by

$$R(k) = \sigma^2 \rho^{|k|} \quad .$$

This form of autocovariance function is commonly encountered in the second-moment analysis of discrete-time random processes. If the linear mean square estimate of X_{n+1} in terms of $X_n, X_{n-1}, X_{n-2}, \ldots$ coincides with the linear mean square estimate of X_{n+1} in terms of X_n, then the sequence X_n is said to be wide sense Markov. It can be shown [7,pp. 420-421] that an equivalent condition for a sequence to be wide sense Markov is for the autocovariance function to be of the form

$$R(k) = \sigma^2 \rho^{|k|} \quad .$$

Obviously, not all Markov sequences are wide sense Markov. In the following theorem we characterize the class of Markov sequences X_n which admit the above form of autocovariance function.

Theorem 3: The second order Markov sequence X_n has an autocovariance function

$$R(k) = \sigma^2 \rho^{|k|}$$

if and only if for no more than one value of n, $a_n \neq 0$ and

$$\int (x-\mu)\theta_n(x) f(x) dx \neq 0.$$

Proof: Let $g(x) = x-\mu$. Then the autocovariance function is given by Eq. (4). The necessary and sufficient condition in the theorem is equivalent to saying that, for at most one value of n, both a_n and b_n in Eq. (4) are nonzero. This, in turn, is equivalent to the autocovariance function being of the specified form. ||

Corollary 1: If, for some n, $\theta_n(x)$ is linear, then the autocovariance function of X_n is given by

$$R(k) = \sigma^2 \rho^{|k|} \quad .$$

Proof: Assume that $\theta_i(x)$ is linear. Then it follows from the orthonormalization that

$$\theta_i(x) = \frac{x-\mu}{\sigma} \quad .$$

Thus we see that

$$\int (x-\mu)\theta_n(x)f(x)dx = \begin{cases} 0 & \text{if } n\neq i \\ \sigma & \text{if } n=i \end{cases}$$

The result follows by Theorem 3. ||

If we are given the bivariate density $h(x,y)$, it may not be obvious as to whether one of the orthonormal functions is linear or not. The following corollary provides a way of checking this.

Corollary 2: If

$$\int h(x,y)(x-\mu)dx = \alpha(y-\mu)f(y) \; ,$$

where α is some constant, then the autocovariance function of X_n is given by

$$R(k) = \sigma^2 \rho^{|k|} \; .$$

Proof: From the hypothesis of the corollary, it follows that the function

$$\left(\frac{x-\mu}{\sigma}\right)\sqrt{f(x)}$$

is an eigenfunction of the operator A. Thus we see that for some n, $\theta_n(x)$ is linear; and the result follows from Corollary 1. ||

In particular, it is seen that if the bivariate density $h(x,y)$ belongs to the class Λ, then the autocovariance function of X_n is of the form

$$R(k) = \sigma^2 \rho^{|k|} \; .$$

When one of the orthonormal functions, say $\theta_i(x)$, is linear, the minimum mean squared error predictor of the sequence X_n takes a particularly simple form. Letting

$$g(x) = \theta_i(x) = \frac{x-\mu}{\sigma} \; ,$$

it follows that

$$b_n = \begin{cases} 1 & \text{if } n=i \\ 0 & \text{if } n\neq i \end{cases}$$

By Eq. (5), the minimum mean squared error estimate of

$$\frac{X_{n+k} - \mu}{\sigma}$$

in terms of X_n, X_{n-1}, X_{n-2},... is given by

$$\rho^k \left(\frac{X_n - \mu}{\sigma}\right) .$$

Therefore the k step minimum mean squared error predictor of the sequence X_n is given by

$$\hat{X}_{n+k} = \rho^k X_n - \mu(\rho^k - 1).$$

IV. RANDOM PULSE TRAINS

Models for random pulse trains are often constructed in the following fashion. Let Y_n be a discrete-time random process and let

$$Z(t) = Y_n, \quad n \le t < n+1.$$

Now let $P(t) = Z(t+\Psi)$, where Ψ is a random variable that is uniformly distributed in [0,1]. The resulting random process P(t) is frequently taken as a model for a random pulse train. Assuming that Y_n is a wide sense stationary process, it is readily seen that P(t) is wide sense stationary and that the autocovariance function $r(\tau)$ is given by $r(\tau) = (|\tau|-k)[R(k+1)-R(k)]+R(k)$ for $|\tau| \in [k,k+1]$, where R(k) is the autocovariance function of the discrete-time random process Y_n. The autocovariance function $r(\tau)$ can be visualized by connecting the adjacent points on the graph of R(k) with straight lines.

The statistics of the random process P(t) are determined by the statistics of the sequence Y_n. Thus we are presented with the problem of modeling the random sequence Y_n. We will assume that the sequence Y_n is equivalent to the sequence X_n, where X_n is the stationary, second-order, Markov sequence that we considered earlier and that was derived from the bivariate density of Eq. (1). A complete statistical description of the random process P(t) can be derived from the bivariate density h(x,y).

We will concentrate on the second moment properties of P(t). It follows from Theorem 2 that either

$$r(\tau) = \begin{cases} \sigma^2(1-|\tau|) & \text{for } |\tau| \le 1 \\ 0 & \text{for } |\tau| > 1, \end{cases}$$

or else $r(\tau)$ will be nonzero almost everywhere on the real line. We will consider the former case.

Lemma 1: Consider the Markov sequence X_n derived from the bivariate density in Eq. (1). If

$$\int (x-\mu) h(x,y)\,dx = 0 \quad \text{for all } y, \qquad (6)$$

then the autocovariance function of the Markov sequence X_n is given by

$$R(k) = \begin{cases} \sigma^2 & \text{for } k=0 \\ 0 & \text{for } k \ne 0 \end{cases} .$$

Proof: The lemma follows from Corollary 2. ||

There are numerous bivariate densities that satisfy Eq. (6), many of which differ radically from each other. Assuming that the bivariate density h(x,y) satisfies Eq. (6), the autocovariance function of the random process P(t) will be given by

$$r(\tau)=\begin{cases}\sigma^2(1-|\tau|) , & |\tau|\le 1\\ 0 , & |\tau|>1.\end{cases}$$

In the above fashion we can construct a family of random pulse trains which share the same autocovariance function. Each random pulse train P(t) is completely determined statistically by the corresponding bivariate density function h(x,y). The bandwidth requirements of each pulse train in this family will be identical, despite the fact that the pulse trains in the family may differ radically from each other. The extension to similar families which share other bandwidth requirements is straightforward.

ACKNOWLEDGEMENTS

This research is supported by the Air Force Office of Scientific Research under Grant AFOSR-71-2101.

REFERENCES

1. H.O. Lancaster, "The Structure of Bivariate Distributions", Ann. Math. Stat., Vol. 29, pp. 719-736, 1958.

2. H.O. Lancaster, "Correlations and Canonical Forms of Bivariate Distributions", Ann. Math. Stat., Vol. 34, pp. 532-538, 1963.

3. J.F. Barrett and D.G. Lampard, "An Expansion for Some Second-Order Probability Distributions and Its Application to Noise Problems", IRE Trans. Information Theory, Vol. IT-1, pp. 10-15, March 1955.

4. F. Oberhettinger, Fourier Expansions A Collection of Formulas, p. 32, #3.11, Academic Press, New York, 1973.

5. A.N. Kolmogorov and S.V. Fomin, Elements of the Theory of Functions and Functional Analysis, Vol. 2, p. 120, Graylock Press, Baltimore, Maryland, 1961.

6. F. Riesz and B. Sz.-Nagy, Functional Analysis, p. 243, Ungar, New York, 1955.

7. A. Papoulis, Probability, Random Variables, and Stochastic Processes, McGraw-Hill, New York, 1965.

8. P. Beckmann, Orthogonal Polynomials for Engineers and Physicists, Appendix I, The Golem Press, Boulder, Colorado, 1973.

9. D.R. Cox and H.D. Miller, The Theory of Stochastic Processes, p. 289, Methuen, London, 1965.

ESTIMATION OF PARAMETERS IN SYSTEMS OF RELATIVELY SLOW TIME VARIATION*

Philip H. Fiske
William L. Root
The University of Michigan, Ann Arbor

ABSTRACT

The estimation of parameters for a rather general class of nonlinear, time-varying, causal, bounded memory systems is discussed. A model for such systems is established using the concept that the system variation is unknown, but bounded between observation times. A recursive estimation algorithm is developed for the model, and both upper and lower bounds are found for the guaranteed error of estimation.

DEVELOPMENT OF THE SYSTEM MODEL

In this paper the term system means simply a mapping from an appropriate space of inputs to an appropriate space of outputs. The central purpose of the paper is to obtain estimates on how rapidly causal systems, of a certain rather general class that includes nonlinear as well as linear systems, can vary with time and still allow adequate experimental identification.

A particular kind of system model is especially appropriate for this study and is used here. It has the features that it is linear in the parameters, even though the system itself may be nonlinear, and that it is widely applicable if the only noise to be considered is output observation noise. We briefly and heuristically describe such a model in a special case in what follows, and this is sufficient for this paper. However there is a general theory of such models, and most of the ad hoc assumptions made below can either be shown to be justifiable or not needed (see [1]).

Let $\mathcal{X}$ be an arbitrary input space, $\mathcal{Y}$ a linear output space, and $\mathcal{H}$ a set of mappings from $\mathcal{X}$ into $\mathcal{Y}$ (i.e., $\mathcal{H}$ is a class of "systems"), so that one may write

$$y = h(x), \quad x \in \mathcal{X}, \quad y \in \mathcal{Y}, \quad h \in \mathcal{H}. \tag{1}$$

Each $x \in \mathcal{X}$ determines a function from $\mathcal{H}$ into $\mathcal{Y}$. If this function is denoted by X, then

$$y = X(h), \quad h \in \mathcal{H}. \tag{2}$$

Let addition and scalar multiplication be defined in $\mathcal{H}$ in the way they usually are for functions and then extend $\mathcal{H}$, if necessary, so that it is closed under linear combinations. Denote this linear extension by $\mathcal{F}$. It is now trivial to verify that the mapping X as extended to $\mathcal{F}$ will be linear, even

---

*Research sponsored by the United States Air Force, Air Force Office of Scientific Research Grant No. 72-2328-B.

though the mapping h may be nonlinear (see e.g., [1]). If additive output noise is present, (1) and (2) are replaced by

$$y = h(x) + v \tag{3}$$

and

$$y = Xh + v, \quad h \in \mathfrak{F}, \tag{4}$$

where v is a random variable taking values in $\mathcal{Y}$ and X is linear. Thus the problem of identifying the system has been put in the form of the classical problem of estimating h in the linear model (4). Of course, equation (4) is abstract, so we now need to specialize it so as to have a meaningful estimation problem. Here we simply assume what we need in order to do this easily.

Suppose that a system with vector-valued inputs and outputs can be adequately represented in terms of sampled inputs and outputs. Futhermore, assume that it has bounded memory (this is the only critical assumption). Then an input is a vector sequence $\{x_k\}$, the corresponding output is a vector sequence $\{y_k\}$, and if the system has memory m,

$$y_k = h_k(x_k, x_{k-1}, \ldots, x_{k-m}) + v_k$$

or

$$y_k = h_k(\underline{x}_k) + v_k, \quad k = 1\ 2, \ldots, \tag{5}$$

where

$$\underline{x}_k \overset{d}{=} (x_k, x_{k-1}, \ldots, x_{k-m}).$$

If the system is time-invariant each h_k is the same; otherwise, of course, the h_k's are different. To each $\underline{x}_k$ there corresponds a linear mapping X_k defined, as before, by $X_k h_k \overset{d}{=} h_k(\underline{x}_k)$, so that

$$y_k = X_k h_k + v_k, \quad k = 1,2,\ldots . \tag{6}$$

Let there be imposed the further condition that all the mappings h_k are fully characterized by a finite set of parameters, say p in number. The $\{h_k\}$ can then be represented by p-vectors, and if the output is an r-vector, the $\{X_k\}$ can be represented by r x p matrices.

If the systems in $\mathfrak{F}$ are of uniformly bounded time variation for each finite interval, one can write

$$h_{k+1} = h_k + w_k, \quad k = 1,2,\ldots, \tag{7}$$

where each component of w_k is bounded. The system model then becomes

$$\begin{cases} h_{k+1} = h_k + w_k \\ y_k = X_k h_k + v_k, \quad k = 1,2,\ldots, \end{cases} \tag{8}$$

where the $\{w_k\}$ are bounded, unknown elements of R^p and the $\{v_k\}$ are random variables representing output noise. The equations (8) are, of course, in the form of the system equations for a degenerate linear dynamical system, but the interpretation given to them is entirely different.

We wish to consider successive blocks of measurements where each block consists of tests with the same suitably chosen N inputs $\{x_1,\ldots,x_N\}$. We make the important simplifying assumption that the system parameters change only between blocks of measurements. Results can be obtained without this assumption (see [3]), but there is not space to go into the additional argument necessary here. This assumption gives an acceptable approximation anyway in many cases where there is only slight variation within the blocks.

With this assumption we have

$$\begin{cases} h^{k+1} = h^k + w^k \\ y^k = Xh^k + v^k, \quad k = 1,2,\ldots, \end{cases} \tag{9}$$

where $h^k \overset{d}{=} h_{kN}$ and

$$y^k = \begin{bmatrix} y_{(k-1)N+1} \\ y_{(k-1)N+2} \\ \vdots \\ y_{kN} \end{bmatrix}, \quad X \overset{d}{=} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_N \end{bmatrix}, \quad v^k = \begin{bmatrix} v_{(k-1)N+1} \\ v_{(k-1)N+2} \\ \vdots \\ v_{kN} \end{bmatrix}.$$

The condition to be imposed on the inputs $\{x_1,\ldots,x_N\}$ is that the operator X has zero null space, so that the vector h^k is estimable. The usual assumptions are made about the noise,

$$Ev^k = 0, \quad Ev^k(v^j)^T = R\delta_{jk}, \tag{10a}$$

where R is strictly definite. The boundedness condition on the w^k is taken to be in the form,

$$|(w^k,\psi_i)| \leq \eta_i, \quad i = 1,\ldots,p, \tag{10b}$$

for all $k = 1,2,\ldots$, where $\{\psi_i\}$ is a complete orthonormal basis of eigenvectors of $(X^TR^{-1}X)^{-1}$, i.e.,

$$(X^TR^{-1}X)^{-1}\psi_i = \sigma_i^2\psi_i, \quad i = 1,\ldots,p.$$

The eigenvalues $\{\sigma_i^2\}$ are real and nonnegative.

ESTIMATION OF PARAMETERS

We now derive a recursive estimation algorithm for the parameter vector h^k and obtain a uniform upper bound on the mean-squared error. Since the actual mean-squared error depends on the $\{w^k\}$, which are unknown, an exact expression for mean-squared error cannot be given, of course.

We begin by forming the standard linear, unbiased, minimum variance (LUMV) estimate of h^1

$$\hat{h}^1 = Cy^1, \tag{11}$$

where $C \overset{d}{=} (X^TR^{-1}X)^{-1}X^TR^{-1}$. The error in this estimate satisfies

$$E\|\hat{h}^1 - h^1\|^2 = \sum_{i=1}^{p} \sigma_i^2 = \sum_{i=1}^{p} b_{1i}^2 \tag{12}$$

where $b_{1i}^2 \overset{d}{=} \sigma_i^2$, $i = 1,\ldots,p$. Now consider the second observation

$$y^2 = Xh^2 + v^2 = Xh^1 + Xw^1 + v^2,$$

which can be rewritten as

$$y^2 - X\hat{h}^1 = X[(h^1 - \hat{h}^1) + w^1] + v^2. \tag{13}$$

Define $z^2 \overset{d}{=} y^2 - X\hat{h}^1$ and $\xi^1 \overset{d}{=} (h^1 - \hat{h}^1) + w^1 = h^2 - \hat{h}^1$ so that (13) becomes

$$z^2 = X\xi^1 + v^2. \tag{14}$$

Let $\mathring{\xi}^1$ be a linear estimate of ξ^1 that is a function of Cz^2, but is otherwise arbitrary. Thus,

$$\mathring{\xi}^1 = \sum_{i,j=1}^{p} a_{ij}(Cz^2,\psi_j)\psi_i \tag{15}$$

where the $\{a_{ij}\}$ are real numbers. The error in this estimate is given by

$$\mathring{\xi}^1 - \xi^1 = \sum_{i=1}^{p}\left[(a_{ii}-1)(\xi^1,\psi_i) + \sum_{\substack{j=1\\(j\neq i)}}^{p} a_{ij}(\xi^1,\psi_j) + \sum_{j=1}^{p} a_{ij}(Cv^2,\psi_j)\right]\psi_i. \tag{16}$$

Therefore, it follows from (10a) and the properties of the $\{\psi_i\}$ that

$$\begin{aligned} E\|\mathring{\xi}^1 - \xi^1\|^2 &= \sum_{i=1}^{p} E\left[(a_{ii}-1)(\xi^1,\psi_i) + \sum_{\substack{j=1\\(j\neq i)}}^{p} a_{ij}(\xi^1,\psi_j)\right]^2 \\ &\quad + \sum_{i,j=1}^{p} a_{ij}^2\sigma_j^2 \leq 2\sum_{i=1}^{p}(a_{ii}-1)^2 E[(\xi^1,\psi_i)^2] \\ &\quad + 2\sum_{i=1}^{p} E\left[\sum_{\substack{j=1\\(j\neq i)}}^{p} a_{ij}(\xi^1,\psi_j)\right]^2 + \sum_{i,j=1}^{p} a_{ij}^2\sigma_j^2 \end{aligned} \tag{17}$$

To minimize the upper bound in (17), one should obviously choose $a_{ij} = 0$ for $i \neq j$. Making this choice yields

$$E\|\mathring{\xi}^1 - \xi^1\|^2 = \sum_{i=1}^{p}(a_{ii}-1)^2 E[(\xi^1,\psi_i)^2] + \sum_{i=1}^{p} a_{ii}^2\sigma_i^2 \tag{18}$$

Now

$$\begin{aligned} E[(\xi^1,\psi_i)^2] &= E[(h^1 - \hat{h}^1,\psi_i)^2] + 2E[(h^1 - \hat{h}^1,\psi_i)(w^1,\psi_i)] + E[(w^1,\psi_i)^2] \\ &\leq b_{1i}^2 + 2b_{1i}\eta_i + \eta_i^2{}^{*} = (b_{1i} + \eta_i)^2, \end{aligned}$$

---

[*] Obviously $E[(h^1 - \hat{h}^1,\psi_i)(w^1,\psi_i)] = 0$, but in general $E[(h^n - \hat{h}^n,\psi_i)(w^n,\psi_i)] \neq 0$, $n \geq 2$.

so

$$E\|\overset{\circ}{\xi}{}^1 - \xi^1\|^2 \leq \sum_{i=1}^{p}\left[(a_{ii} - 1)^2(b_{1i} + \eta_i)^2 + a_{ii}^2\sigma_i^2\right]. \tag{19}$$

The upper bound in (19) is minimized by putting

$$a_{ii} = \frac{(b_{1i} + \eta_i)^2}{(b_{1i} + \eta_i)^2 + \sigma_i^2}, \quad i = 1,\ldots,p.$$

The linear estimate that results when these values are used is

$$\hat{\xi}^1 = \sum_{i=1}^{p} \frac{(b_{1i} + \eta_i)^2}{(b_{1i} + \eta_i)^2 + \sigma_i^2}(Cz^2,\psi_i)\psi_i. \tag{20}$$

Since $h^2 = \hat{h}^1 + \xi^1$, the estimate of h^2 is taken to be

$$\hat{h}^2 = \hat{h}^1 + \hat{\xi}^1. \tag{21}$$

Then

$$\begin{aligned} E\|\hat{h}^2 - h^2\|^2 &= E\|(\hat{h}^1 + \hat{\xi}^1) - (\hat{h}^1 + \xi^1)\|^2 = E\|\hat{\xi}^1 - \xi^1\|^2 \\ &\leq \sum_{i=1}^{p}\left\{\left[\frac{(b_{1i} + \eta_i)^2}{(b_{1i} + \eta_i)^2 + \sigma_i^2} - 1\right]^2(b_{1i} + \eta_i)^2\right. \\ &\quad \left. + \left[\frac{(b_{1i} + \eta_i)^2}{(b_{1i} + \eta_i)^2 + \sigma_i^2}\right]^2\sigma_i^2\right\} = \sum_{i=1}^{p}\frac{(b_{1i} + \eta_i)^2\sigma_i^2}{(b_{1i} + \eta_i)^2 + \sigma_i^2} = \sum_{i=1}^{p} b_{2i}^2 \end{aligned} \tag{22}$$

where $\quad b_{2i}^2 \overset{d}{=} \dfrac{(b_{1i} + \eta_i)^2\sigma_i^2}{(b_{1i} + \eta_i)^2 + \sigma_i^2}, \quad i = 1,\ldots,p.$

If this process is continually repeated, we obtain the estimation algorithm

$$\begin{aligned} \hat{h}^k &= \hat{h}^{k-1} + \hat{\xi}^{k-1} \\ \hat{\xi}^{k-1} &= \sum_{i=1}^{p}\frac{(b_{(k-1)i} + \eta_i)^2\sigma_i^2}{(b_{(k-1)i} + \eta_i)^2 + \sigma_i^2}(Cy^k - \hat{h}^{k-1},\psi_i)\psi_i \\ b_{ki}^2 &= \frac{(b_{(k-1)i} + \eta_i)^2\sigma_i^2}{(b_{(k-1)i} + \eta_i)^2 + \sigma_i^2}, \quad i = 1,\ldots,p. \end{aligned} \tag{23}$$

The initial conditions are given by

$$\hat{h}^1 = Cy^1, \quad b_{1i}^2 = \sigma_i^2, \quad i = 1,\ldots,p \tag{24}$$

and the error in the estimate of h^k satisfies

$$E\|\hat{h}^k - h^k\|^2 \leq \sum_{i=1}^{p} b_{ki}^2 \tag{25}$$

Lemma 1. If $f(x)$ is twice continuously differentiable for $x > 0$ and satisfies

(i) $\quad \lim\limits_{x\to 0+} f(x) > 0$

(ii) $\lim_{x \to \infty} f(x) < \infty$

(iii) $\frac{df(x)}{dx} > 0, \; x > 0$

(iv) $\frac{d^2 f(x)}{dx^2} < 0, \; x > 0$

then (a) $f(x) = x$ has a unique solution x^* for $x > 0$.

(b) the sequence $x_n = f(x_{n-1})$, $x_1 > 0$, converges to x^*.

Proof: Follows from elementary arguments. ||| The limiting upper bound can now be established.

Proposition 1. If x_i^* is the solution of the equation

$$\frac{(x^{1/2} + \eta_i)^2 \sigma_i^2}{(x^{1/2} + \eta_i)^2 + \sigma_i^2} = x,$$

then $\lim_{n \to \infty} b_{ni}^2 = x_i^*$.

Proof: Define

$$\phi(x) \overset{d}{=} \frac{(x^{1/2} + \eta_i)^2 \sigma_i^2}{(x^{1/2} + \eta_i)^2 + \sigma_i^2}$$

and note that $\phi(b_{(n-1)i}^2) = b_{ni}^2$. Result follows directly from application of Lemma 1. ||| Hence, use of the estimation algorithm in (23) yields estimates $\{\hat{h}^k\}$ satisfying $\lim_{k \to \infty} E\|\hat{h}^k - h^k\|^2 \leq \sum_{i=1}^{p} x_i^*$.

LOWER BOUNDS ON THE ESTIMATION ERROR

We now derive a lower bound on the attainable mean-squared error, in the sense that with any linear estimator the error can always be as large as this bound if certain parameter changes occur. Of course, in many cases the actual error may be less than this lower bound, but we can never "guarantee" this fact. The basic idea is to construct a problem in which the system variations are random, with known mean and covariance, but bounded as in (10b). Kalman filtering techniques can then be applied in order to find the minimum mean-squared error which is attainable using any linear estimation procedure. This error provides the lower bound of interest.

Suppose that the w^k's are random vectors, independent of the v^k's, which satisfy (10b) and are described by

$$P[(w^k, \psi_i) = \eta_i] = \frac{1}{2} = P[(w^k, \psi_i) = -\eta_i], \tag{26}$$

and

$$E[(w^k, \psi_i)(w^\ell, \psi_j)] = \eta_i^2 \delta_{ij} \delta_{k\ell}.$$

Then

$$Ew^k = 0, \quad Ew^k (w^j)^T = Q\delta_{kj}, \tag{27}$$

where Q can be found from knowledge of the $\{\psi_i\}$ and (26). Suppose also that h^1 is a random vector with, say, mean zero and covariance matrix P_1 =

$(X^T R^{-1} X)^{-1}$. The error covariance matrix for the Kalman estimate $\hat{h}^k$ of h^k is given by the recursive equation (see e.g., [4])

$$P_k = (P_{k-1} + Q)\{I - X^T[X(P_{k-1} + Q)X^T + R]^{-1} \cdot \\ \cdot X(P_{k-1} + Q)\}, \quad k = 2,3,\ldots . \tag{29}$$

We then have:

Proposition 2. The upper bound on the attainable mean-squared error of any linear estimate of the system parameters $\{h^k\}$, in the model described by (9) and (10), cannot be less than $\inf_k \operatorname{Tr} P_k \overset{d}{=} e$, where P_k $(k \geq 2)$ is given by (29) and $P_1 = (X^T R^{-1} X)^{-1}$.

Proof: Suppose the $\{w^k\}$ are random vectors satisfying (26) and h^1 is a random vector with mean zero and covariance P_1. First we observe that if only y^1 is available, there is no linear estimator that will yield uniformly smaller mean-squared error than P_1. This is because P_1 was chosen to be the variance of the LUMV estimate for h^1 in the equation $y^1 = Xh^1 + v^1$, and no biased estimate gives a mean-squared error bounded for all h^1. Next, suppose that for $m \geq 2$ there exists a linear estimator $\hat{\hat{h}}^m$ such that

$$E\{\|\hat{h}^m - h^m\|^2 \mid w^1,\ldots,w^{m-1}\} < e$$

for every sequence $\{w^1,\ldots,w^{m-1}\}$. Then

$$E\|\hat{h}^m - h^m\|^2 = E\{E\{\|\hat{h}^m = h^m\|^2 \mid w^1,\ldots,w^{m-1}\}\} < e.$$

But this contradicts the minimum mean-squared error property of the Kalman estimate $\hat{h}^m$. Consequently, for any m, there is a sequence $w^1,\ldots,w^{m-1}$ such that $E\{\|\hat{\hat{h}}^m - h^m\|^2 \mid w^1,\ldots,w^{m-1}\} \geq e$. Since $\{w^1,\ldots,w^{m-1}\}$ satisfies (10b), this proves the assertion. |||

It is of interest to interpret this result in the special case of (9) and (10) where X is the identity and R is diagonal. The eigenvectors $\{\psi_i\}$ then become the standard unit vectors in R^p and $Q = \operatorname{diag}(\eta_1^2,\eta_2^2,\ldots,\eta_p^2)$. Also, with a little algebraic manipulation (29) becomes

$$P_k = (P_{k-1} + Q)(P_{k-1} + Q + R)^{-1} R.$$

Since $P_1 = R$ is diagonal, all the P_k will be diagonal. In particular, if $P_{k-1} = \operatorname{diag}(\alpha^2_{(k-1)1},\alpha^2_{(k-1)2},\ldots,\alpha^2_{(k-1)p})$ then $P_k = \operatorname{diag}(\alpha^2_{k1},\alpha^2_{k2},\ldots,\alpha^2_{kp})$ where

$$\alpha^2_{ki} = \frac{(\alpha^2_{(k-1)i} + \eta_i^2)\sigma_i^2}{\alpha^2_{(k-1)i} + \eta_i^2 + \sigma_i^2}, \quad i = 1,\ldots,p. \tag{30}$$

Corollary. If for the system described by (9) and (10), X is the identity and R is diagonal, then there exists a sequence $\{w^k\}$ satisfying (10b) such that

$$E\|\hat{h}^k - h^k\|^2 \geq \sum_{i=1}^{p} \left\{ -\frac{\eta_i^2}{2} + \frac{\eta_i^2}{2}\sqrt{1 + 4\frac{\sigma_i^2}{\eta_i^2}} \right\}$$

for any linear estimate $\hat{h}^k$ of h^k.

Proof: Define

$$\theta(x) \overset{d}{=} \frac{(x+\eta_i^2)\sigma_i^2}{x+\eta_i^2+\sigma_i^2}$$

and note that for equation (30), $\theta(\alpha^2_{(k-1)i}) = \alpha^2_{ki}$. Application of Lemma 1 yields

$$\lim_{k\to\infty} \alpha^2_{ki} = -\frac{\eta_i^2}{2} + \frac{\eta_i^2}{2}\sqrt{1+4\frac{\sigma_i^2}{\eta_i^2}}, \quad i = 1,\ldots,p.$$

Since

$$\inf_k \operatorname{Tr}P_k = \lim_{k\to\infty} \operatorname{Tr} P_k = \lim_{k\to\infty} \sum_{i=1}^{p} \alpha^2_{ki} = \sum_{i=1}^{p}\left\{-\frac{\eta_i^2}{2} + \frac{\eta_i^2}{2}\sqrt{1+4\frac{\sigma_i^2}{\eta_i^2}}\right\},$$

the result follows directly from Proposition 2. ||| Figure 1 provides a plot of the i^{th} component of the upper and lower bounds on the mean-square error when $X = I$ and R is diagonal.

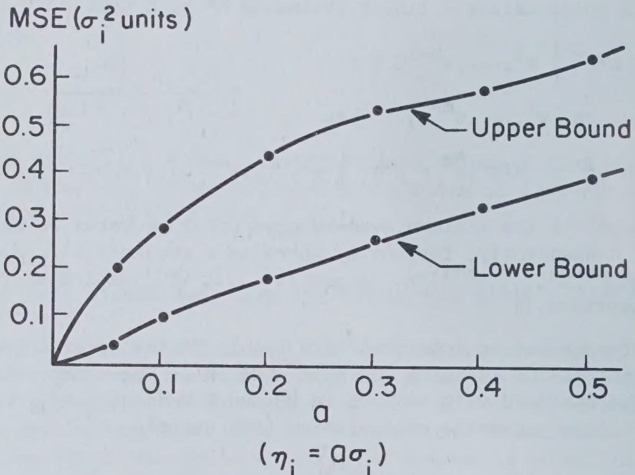


Figure 1. Bounds on Errors

COMMENTS

(1) The results derived for estimating the parameters when they vary only between blocks of measurements can be extended to include the case where the parameters vary between each pair of measurements (see [3]).

(2) The upper and lower bound expressions given by Propositions 1 and 2 provide criteria for determining whether or not a system can be considered as being slowly varying (see [3]).

(3) The restriction to the case where X is the identity is in fact less restrictive than it would appear to be at first glance. The "standard

ε-representation" developed in [1] does provide such a model for a very large class of systems. Of course, the parametrization provided by this model may not be what is desired in a practical situation. However, our point of view as regards to the concept of slowly varying is that if there is any parametrization for which sufficiently good estimates can be made, then the system is slowly varying.

(4) The estimate used at each state of the recursive estimation procedure is essentially the modified LUMV estimate described in [2].

REFERENCES

[1] W. L. Root, "On the Modelling of Systems for Identification. Part 1. ε-Representations of Classes of Systems. Part 2. Time-Varying Systems," to appear in SIAM J. Control.

[2] W. L. Root, "On the Modelling and Estimation of Communication Channels," Multivariate Analysis III, ed. Krishnaiah, Academic Press, 1973.

[3] P. H. Fiske and W. L. Root, "On the Modelling and Identification of Slowly Varying Systems," Technical Report, The University of Michigan, College of Engineering (in preparation).

[4] K. J. Åström, Introduction to Stochastic Control Theory, Academic Press, 1970.

MINIMAX PARAMETER ESTIMATION UNDER GENERALIZED QUADRATIC LOSS WITH A COMPACT PARAMETER SPACE

R. PADILLA
Department of Electrical Engineering
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois 61801

M. MINTZ
Department of Systems Engineering
Moore School of Electrical Engineering
University of Pennsylvania
Philadelphia, Pennsylvania

ABSTRACT

A linear minimax estimate for the mean of a normal random vector with known covariance is derived for a loss function which is quadratic in both the estimation error and the unknown mean; the mean is restricted to lie in a known hyperellipsoidal subset of E^n. The derivation is based on the method of least favorable prior distributions. The structure of the solution is determined by the solution to a certain nonlinear algebraic matrix equation.

1. INTRODUCTION

The estimation problem we shall consider in this paper belongs to a general class of problems that can be described by the following model:

Let $z = \theta + v$ where $\theta \in \Omega$, Ω is some specified parameter space contained in E^n, and v is a zero-mean Gaussian random vector with known positive definite covariance matrix Σ. We want to find a minimax estimator for the unknown parameter θ. It is known [2,3,4] that the existence of minimax estimators and the structure of these estimators when they exist is strongly dependent on the structure of the problem which is affected by the underlying loss function, the parameter space Ω and the class of allowed decision rules.

When the loss function is the Standard Quadratic Loss function,

$$L_0[\delta(z),\theta] = [\delta(z) - \theta]'C[\delta(z) - \theta] , \tag{1.1}$$

and the parameter space Ω is E^n, a minimax solution for this problem is $\delta(z) = z$; see [1], p. 170 for further details.

The standard quadratic loss function is generalized by adding to it a quadratic term in θ obtaining

$$L_1[\delta(z),\theta] = [\delta(z) - \theta]'C[\delta(z) - \theta] - \theta'D\theta . \tag{1.2}$$

This loss function forces Nature to compromise between a gain due to increasing the Statistician's estimation error and a loss which is equal to the square of the norm of θ weighted by D. For an unconstrained parameter space, i.e., $\Omega = E^n$, this problem was solved by Basar and Mintz [2].

The parameter θ can be restricted to reside within a known compact subset of E^n, for example a hyperellipsoid defined by

$$\Omega = \{\theta \in E^n : \theta'G\theta \le e^2,\ e^2 > 0,\ G > \underline{0}\}. \tag{1.3}$$

In this case a linear minimax estimator for θ under the standard quadratic loss function was found by Mintz [3]. We will be considering here the problem of finding a linear minimax estimator under a generalized quadratic loss function and will constrain the parameter θ to reside within a known hyperellipsoid in E^n.

2. PROBLEM STATEMENT

Let $z = \theta + v$ where v is a normal zero-mean random vector with known positive definite covariance matrix Σ and θ is an unknown element of a known hyperellipsoidal subset of E^n denoted by Ω and defined by

$$\Omega = \{\theta \in E^n : \theta' D_2 \theta \le e^2 \;;\; e^2 > 0,\ D_2 > \underline{0}\} \tag{2.1}$$

We wish to determine a minimax estimate $\delta(z)$ for θ with respect to the generalized quadratic loss function

$$L[\delta(z),\theta] = [\delta(z) - \theta]' C [\delta(z) - \theta] - \theta' D_1 \theta \tag{2.2}$$

where $C \ge \underline{0}$ and $D_1 > \underline{0}$, subject to the restriction that δ be a linear rule.

3. SOLUTION

Define

$$\Gamma(\Lambda,\alpha) \triangleq (\Lambda + \Sigma)^{-1} \Sigma\, C\, \Sigma (\Lambda + \Sigma)^{-1} - D_1 - \alpha D_2 \tag{3.1}$$

with

$$\Lambda \ge \underline{0} \quad \text{and} \quad \alpha \ge 0 \tag{3.2}$$

Let Λ^* and α^* denote any solution to the matrix equation:

$$\Gamma(\Lambda^*,\alpha^*)\Lambda^* = \underline{0} \tag{3.3}$$

such that

$$\Gamma(\Lambda^*,\alpha^*) \le \underline{0} \tag{3.4}$$

and

$$\mathrm{Tr}[D_2 \Lambda^*] \le e^2. \tag{3.5}$$

Theorem I:

(i) If $\Gamma(\Lambda,\alpha)$ is defined by (3.1) and (3.2) then there exists a covariance matrix Λ^* and a scalar $\alpha^* \ge 0$ that satisfy (3.3) and (3.4), and

either (a) $\alpha^* = 0$ and $\mathrm{Tr}[D_2\Lambda^*] = e_*^2 \le e^2$

or (b) $\alpha^* > 0$ and $\mathrm{Tr}[D_2\Lambda^*] = e_*^2 = e^2$.

(ii) If $\delta(z)$ is defined by

$$\delta^*(z) = \Lambda^*(\Lambda^* + \Sigma)^{-1} z \tag{3.6}$$

then $\delta^*(z)$ is a linear minimax estimate for θ. A proof of part (i) of the main theorem may be obtained by generalizing the proof that appears in Appendix I of [3]. The essential modification of that result needed to show part (i) is obtained by replacing $J(A,\Lambda)$ as defined by Lemma 2 of [3] by

$$J(A,\Lambda) \triangleq \mathrm{Tr}\,\{[(I - A)'C\,(I - A) - D_1]\} + \mathrm{Tr}[CA\,\Sigma\, A']. \tag{3.7}$$

We note at this point that the matrix D_2 of this present paper corresponds to D of reference [3]. Since the modified proof is very similar a complete discussion will be omitted. We also note that the last two assertions in Appendix I of [3] do not apply in the present problem due to the difference in the structure of the loss function.

A proof of part (ii) of Theorem I is obtained using the following lemma.

Lemma 1:

Let δ_λ be Bayes with respect to the conditional risk function $r[\delta,\theta]$ and a prior distribution $\lambda(\theta)$ for θ. Assume that $\max\limits_{\theta \in \Omega} r[\delta_\lambda,\theta] = \bar{r}$ exists. Define $\tau_\lambda = \{\theta \in \Omega : r[\delta_\lambda,\theta] = \bar{r}\}$. Assume that $\lambda[\tau_\lambda] = 1$. We now have:

(i) δ_λ is minimax,

(ii) λ is a least favorable prior distribution.

For a proof of this result see [1].

To apply this lemma we need a prior probability distribution for θ which we define as follows:

Assume $\mathrm{Tr}[D_2\Lambda] = e_*^2 \le e^2$, where $\Lambda \ge \underline{0}$ and consider the matrix $D_2^{\frac{1}{2}}\,\Lambda\, D_2^{\frac{1}{2}}$, where $D_2^{\frac{1}{2}}$ is the unique symmetric positive definite square root of D_2. Assume T diagonalizes $D_2^{\frac{1}{2}}\,\Lambda\, D_2^{\frac{1}{2}}$, that is

$$S = T'D_2^{\frac{1}{2}}\,\Lambda\, D_2^{\frac{1}{2}}\, T = \mathrm{diag}\,(s_1, \ldots, s_n). \tag{3.8}$$

Let m denote an n-dimensional random vector with independent components denoted by m_i, $i = 1, \ldots, n$ whose distribution is defined by:

$$\Pr\left\{m_i = s_1^{\frac{1}{2}}\right\} = \frac{1}{2}\ , \quad \Pr\left\{m_i = -\left(s_i^{\frac{1}{2}}\right)\right\} = \frac{1}{2}\,. \tag{3.9}$$

The random vector m takes on 2^k values with equal probability, where k is the rank of Λ. Let $\mu[\Lambda]$ denote the probability measure defined by the distribution of the random vector $y \triangleq D_2^{-\frac{1}{2}}\,Tm$. We note that the support of the measure $\mu[\Lambda]$ lies entirely in the boundary of Ω^*, a hyperellipsoidal subset of E^n contained in Ω and defined by

$$\Omega^* = \left\{y \in E^n : yD_2y \le e_*^2 \le e^2\right\}. \tag{3.10}$$

This assertion follows from the fact that for each of the 2^k equally likely values of m, the vector $y = D_2^{-\frac{1}{2}}\,Tm$ lies on the boundary of Ω^* since

$$y'Dy = m'T'D_2^{-\frac{1}{2}}D_2^{-\frac{1}{2}} Tm = m'm$$

$$= \sum_{i=1}^{n} s_i = \mathrm{Tr}\left[D_2^{\frac{1}{2}} \Lambda D_2^{\frac{1}{2}}\right] = \mathrm{Tr}\left[D_2\Lambda\right] = e_*^2 \leq e^2 . \tag{3.11}$$

Further, we note that $E[y] = 0$ and

$$E[yy'] = E\left[D_2^{-\frac{1}{2}} T m m' T'D_2^{-\frac{1}{2}}\right] = D_2^{-\frac{1}{2}} T S T'D_2^{-\frac{1}{2}}$$

$$= D_2^{-\frac{1}{2}}T T'D_2^{\frac{1}{2}} \Lambda D_2^{\frac{1}{2}} T'T D_2^{-\frac{1}{2}}$$

$$= \Lambda . \tag{3.12}$$

Now, let λ_o denote a prior probability distribution for θ such that $\theta \sim \mu[\Lambda]$. For this prior distribution a Bayes estimate is given by

$$\delta_{\lambda_o}(z) = \Lambda(\Lambda + \Sigma)^{-1}z. \tag{3.13}$$

This follows from the fact that the loss function is quadratic in the estimation error and therefore within the class of linear rules for a prior distribution with covariance Λ the rule defined by (3.13) minimizes the average risk. In what follows we will show that $\lambda^* = \mu[\Lambda^*]$ and $\delta^*(z)$ defined by (3.6) satisfy the conditions stated in Lemma 1 of this section and thus establish part (ii) of the main theorem. We need an expression for the conditional risk function:

$$r\left[\delta_{\lambda_o},\theta\right] = E\left\{L\left[\delta_{\lambda_o}(z),\theta\right]\middle|\theta\right\}$$

$$= \theta'\left[(\Lambda + \Sigma)^{-1} \Sigma C \Sigma(\Lambda + \Sigma)^{-1} - D_1\right] \theta$$

$$+ \mathrm{Tr}[(\Lambda + \Sigma)^{-1} \Lambda C \Lambda (\Lambda + \Sigma)^{-1} \Sigma]. \tag{3.14}$$

The fact that λ^* and $\delta^*(z)$ satisfy Lemma 1 of Section 3 is established in several steps as follows:

Step 1: Observe that for all δ of the form $\delta(z) = \Lambda(\Lambda + \Sigma)^{-1}z$, $\Lambda \geq \underline{0}$ the maximum of the conditional risk function

$$\bar{r} = \max_{\theta \in \Omega} r[\delta,\theta]$$

exists since r is continuous in θ and Ω is compact.

Step 2: First consider the case when the solution to (3.3) and (3.4) satisfy condition (a) of part (i) of the theorem. In this case we note that the conditional risk function can be written as:

$$r[\delta^*,\theta] = \theta'\Gamma(\Lambda^*,0)\theta + \mathrm{Tr}[(\Lambda^* + \Sigma)^{-1} \Lambda^*C \Lambda^*(\Lambda^* + \Sigma)^{-1} \Sigma], \tag{3.15}$$

then (3.4) implies that $\tau_{\lambda}*$, defined by

$$\tau_{\lambda}* = \{ \theta \in \Omega : r[\delta^*,\theta] = \bar{r} \},$$

coincides with the nullspace of $\Gamma(\Lambda^*,0)$, that is

$$\tau_{\lambda}* = N[\Gamma(\Lambda^*,0)] . \tag{3.16}$$

When case (b) of part (i) is in force the conditional risk function can be written as

$$r[\delta^*,\theta] = \theta'\Gamma(\Lambda^*,\alpha^*)\theta + \alpha^*\theta'D\,\theta + \mathrm{Tr}[(\Lambda^* + \Sigma)^{-1}\Lambda^* C\,\Lambda^*(\Lambda^* + \Sigma)^{-1}\Sigma] . \tag{3.17}$$

It is obvious that for this case

$$\tau_{\lambda *} = N[\Gamma(\Lambda^*,\alpha^*)] \cap \partial\,\Omega \tag{3.18}$$

where $\partial\,\Omega = \{\theta \in E^n : \theta' D_2 \theta = e^2\}$. We next prove that in cases (a) and (b) $\lambda^*[\tau_{\lambda *}] = 1$.

Step 3: In both cases considered in step 2, $\tau_{\lambda *}$ is a subset of the nullspace of $\Gamma(\Lambda^*,\alpha^*)$. We will now prove that λ^* assigns probability one to $N[\Gamma(\Lambda^*,\alpha^*)]$. From (3.4) we conclude that $\theta'\Gamma(\Lambda^*,\alpha^*)\theta \le 0$ but

$$\begin{aligned} E\{\theta'\Gamma(\Lambda^*,\alpha^*)\theta\} &= E\{\mathrm{Tr}[\theta'\Gamma(\Lambda^*,\alpha^*)\theta]\} \\ &= E\{\mathrm{Tr}[\Gamma(\Lambda^*,\alpha^*)\theta\theta']\} \\ &= \mathrm{Tr}[\Gamma(\Lambda^*,\alpha^*)\Lambda^*] \end{aligned} \tag{3.19}$$

and (3.3) implies that

$$\mathrm{Tr}[\Gamma(\Lambda^*,\alpha^*)\Lambda^*] = 0. \tag{3.20}$$

So we have

$$\theta'\Gamma(\Lambda^*,\alpha^*)\theta \le 0 \tag{3.21a}$$

and

$$E[\theta'\Gamma(\Lambda^*,\alpha^*)\theta] = 0. \tag{3.21b}$$

This allows us to conclude that λ^* assigns probability one to the nullspace of $\Gamma(\Lambda^*,\alpha^*)$. For the case where Λ^* and α^* satisfy (a) in part (i) this proves that $\lambda^*[\tau_{\lambda *}] = 1$.

For the situation in which case (b) of part (i) is in force, we note that the discussion following (3.10) establishes that $\lambda^* = \mu(\Lambda^*)$ assigns probability one to the boundary of Ω^*, which coincides with Ω in this situation since $\mathrm{Tr}[D_2\Lambda^*] = e^2$. Hence

$$\lambda^*[\partial\Omega] = 1 \quad \text{and} \quad \lambda^*[N(\Gamma(\Lambda^*,\alpha^*))] = 1$$

and therefore we conclude that $\tau_{\lambda*}$ as defined by (3.18) receives probability one from $\lambda*$. We note at this point that we have a prior probability distribution $\lambda*$, a Bayes estimator $\delta*(z)$ with respect to $\lambda*$ and $r[\delta,\theta]$ and $\lambda*$ assigns probability one to the set $\tau_{\lambda*}$ where $r[\delta*,\theta]$ attains its maximum value, i.e., $\bar{r}$. Making use of Lemma 1 of this section we conclude that

$$\delta* = \Lambda*(\Lambda + \Sigma)^{-1}z$$

is a linear minimax estimate for θ. This finishes the proof of part (ii) of the main theorem.

4. REMARKS

The main theorem in Section 3 provides us with a linear minimax estimate of the unknown parameter θ with respect to the generalized quadratic loss function defined in Section 2 where the parameter space is defined to be a hyperellipsoid Ω subset of E^n. The least favorable prior distribution which we associated with this minimax estimate is a discrete probability distribution that has support on the boundary of a hyperellipsoidal subset of E^n, which we denoted by $\Omega*$, contained in the original hyperellipsoid Ω. The minimax estimator, the least favorable prior distribution, and the hyperellipsoid $\Omega*$ are all determined by the solution to the algebraic matrix equations stated in part (i) of Theorem I.

When the restriction imposed on the parameter that it reside in this hyperellipsoid is lifted, Basar and Mintz [2] showed that the minimax estimate, over linear and nonlinear rules, is a linear estimate and the least favorable prior distribution is a zero-mean normal distribution. A closely related problem solved by Mintz [3] in which the loss function is the standard quadratic loss function and the parameter space is a hyperellipsoidal subset of E^n exhibits a similar structure in the least favorable prior distribution to the problem treated here, with the difference that in that case the support of the discrete prior distribution lies always on the boundary of the parameter space. The class of decision rules considered in [3] is also restricted to linear rules.

5. A GENERALIZED MODEL

We consider now a more general observation model. Let $z = x + \theta + v$, where x and v are independent zero-mean Gaussian random vectors with positive definite covariance matrices Q and Σ respectively. The results obtained in Section 3 can be extended to find a linear minimax estimate for $u \triangleq x + \theta$ under a generalized quadratic loss function given by

$$L[\delta(z),u] = (\delta-u)'c(\delta-u) - \theta'D_1\theta.$$

The unknown parameter θ is an element of $\Omega = \{\theta \in E^n : \theta' D_2 \theta \le e^2,\ e^2 > 0\}$. The following theorem provides the desired minimax estimate.

Theorem II:

Define

$$\Gamma(\Lambda,\alpha) \triangleq (\Lambda + Q + \Sigma)^{-1} \Sigma C \Sigma (\Lambda + Q + \Sigma)^{-1} - D_1 - \alpha D_2. \tag{5.1}$$

Let Λ^* and α^* denote any solution to

$$\Gamma(\Lambda^*,\alpha^*)\Lambda^* = 0 \tag{5.2}$$

$$\Gamma(\Lambda^*,\alpha^*) \le 0 \tag{5.3}$$

$$\mathrm{Tr}(D_2\Lambda^*) \le e^2 \tag{5.4}$$

where

$$\Lambda^* \ge \underline{0} \quad \text{and} \quad \alpha^* \ge 0. \tag{5.5}$$

With this definition and notation we conclude that:

(i) There exists a pair Λ^* and α^* satisfying (5.2) through (5.5), and either

$$\alpha^* = 0 \quad \text{and } \mathrm{Tr}[D_2\Lambda^*] = e_*^2 \le e^2$$

or

$$\alpha^* > 0 \quad \text{and } \mathrm{Tr}[D_2\Lambda^*] = e_*^2 = e^2 .$$

(ii) The decision rule δ^* given by

$$\delta^*(z) = (\Lambda^* + Q)(\Lambda^* + Q + \Sigma)^{-1} z \tag{5.6}$$

is a linear minimax estimate for u. The details of the proof are omitted here since it is essentially the same as the proof of Theorem I of Section 3.

A further extension of the work presented in this paper can be used to determine a linear minimax terminal state estimate for a linear plant subject to an energy constraint with a generalized quadratic loss function as the underlying loss function.

6. ACKNOWLEDGEMENT

This work was supported in part by the Air Force Office of Scientific Research under Grant AFOSR 73-2570 and in part by the Joint Services Electronics Program (U.S. Army, U.S. Navy, U.S. Air Force) under Contract DAAB-07-72-C-0259.

7. REFERENCES

1. T. S. Ferguson, Mathematical Statistics - A Decision Theoretic Approach Academic Press, New York, N. Y., 1967.

2. T. Basar and M. Mintz, "On a Minimax Estimate for the Mean of a Normal Random Vector Under a Generalized Quadratic Loss Function," The Annals of Statistics, Vol. 1, No. 1, pp. 127-134, 1973.
3. M. Mintz, "On a Minimax Parameter Estimation Problem with a Compact Parameter Space," Proceedings of the 1972 IEEE Conference on Decision and Control, pp. 178-182.
4. D. J. Stelte and M. Mintz, "On Minimax Parameter Estimation under Exponentiated Generalized Quadratic Loss," Proceedings of the Eleventh Allerton Conference on Circuit and System Theory, Univ. of Ill., pp. 215-225, October 1973.

INDEX OF AUTHORS

Aggarwal, J. K. 862

Aggarwal, Vijay B. 96

Aiken, Richard C. 795

Ashjaee, M. J. 238

Atherton, D. P. 351

Atkins, D. E. 628

Baillieul, John 823

Bandler, J. W. 912, 992

Barnes, Earl R. 143

Barnes, Richard C. 288

Basar, Tamer 199

Baugh, C. R. 535, 542

Berger, Roger L. 469

Bertsekas, Dimitri P. 41

Bhargava, B. K. 278

Blake, Ian F. 782, 787

Branin, F. H., Jr. 703

Brown, David P. 948

Budihardjo, P. S. 759

Burke, V. 975

Bushnell, W. J. 1016

Butvila, Jonas 361

Cha, C. 244

Chan, W. S. 63

Chang, K. W. 790

Charlambous, Christakis 902

Chau, T. H. 942

Cheung, Lawrence 670

Chin, F. 120

Chu, W. Y. 912

Chyung, Dong Hak 940

Cohen, Arthur I. 411

Cook, George E. 720

Cornell, R. G. 649

Cottafava, G. 137

Cullum, Jane 35

D'Antona, O. 137

Dao, Tich T. 557

Davidson, Edward S. 658

Davison, Edward J. 53

Desoer, C. A. 63

Donath, W. E. 35

Druffel, Larry E. 106

Druzeta, A. 547

Dudkikar, Kishan L. 490

DuCasse, Edgar 254, 263

Duffin, R. J. 975

El-Geneidy, Medhat M. 469

Elliott, David L. 844

Eyman, E. D. 932

Feick, Rodolfo 207

Ferrari, Domenico 641

Feuer, Albert 25

Fiske, Philip H. 1039

Foo, Norman Y-K 369
Francis, B. A. 457
Fu, K. S. 278
Gabriel, R. P. 875
Gala, B. E. 298
Garduno, Gilbert 991
Gaschnig, John 866
Gaudiano, F. F. 957
Gault, J. W. 217
Goldstein, A. A. 38
Gressang, R. V. 333
Griffith, Belver C. 512
Grosky, William I. 268
Gruhl, James 599
Gupta, Naresh N. 344
Hanson, Donald A. 887
Hazony, D. 975
Held, Richard E. 1003
Hillman, Donald J. 495
Hoffman, Charles W. 317
Hu, T. C. 393
Hurst, S. L. 525
Ionescu, Tudor 479
Irrgang, Michael E. 307
Iwens, R. P. 180
Jai, A. 217
Jameson, Antony 803
Jamshidi, M. 379
Jones, Vincent C. 858
Kalaycioglu, U. 628
Kaliski, Martin E. 739
Kassam, Saleem A. 1007
Kerntopf, Pawel 115
Kislovski, A. S. 997
Kleinman, Allan J. 588
Kleinman, David L. 588
Klessig, R. 421
Kline, R. M. 730
Konomi, Munetaka 705
Krener, Arthur J. 834
Kurz, L. 1016
Kyriakopoulos, Nicholas 991
Lakshmikantham, V. 71
Lal, M. 967
Lapidus, Leon 795
Lee, S. C. 431
Lee, Ting N. 948
Lemarechal, Claude 16
Lenard, M. L. 393
Liang, Johnnie C. 185
Liu, P. C. 922
Liu, Philip S. 678
Maki, Gary K. 224
Mangasarian, O. L. 153
Marcus, Steven I. 849
Mark, J. W. 759
Martin, David N. 849
Masson, Gerald M. 130

McElroy, T. T. 180
Mehra, R. K. 324
Metze, G. A. 244
Meyer, R. R. 396
Michailesco, G. 809
Michalopoulos, D. A. 298
Michalski, R. S. 885
Michel, Anthony N. 77
Mintz, M. 190, 1048
Mohanty, N. C. 212, 775
Mohler, R. R. 814
Monopoli, R. V. 479
Monzingo, R. A. 171
Mowle, Frederic J. 670, 678
Mukai, H. 161
Muller, David E. 96
Murata, Tadao 982
Nakhla, M. S. 703, 713
O'Malley, R. E., Jr. 803
Osborne, O. D. 897
Ozguner, U. 441
Padilla, C. 190
Padilla, R. 1048
Papantoni-Kazakos, P. 1024
Paz, I. 1024
Peiser, R. R. 431
Perkins, W. R. 441
Polak, E. 161
Preece, Scott E. 502
Priemer, Roland 361
Pullen, Keats A., Jr. 317
Pursley, Michael B. 765
Rabbat, N. B. 730
Rahman, M. 782
Rao, H. S. 580
Rasmussen, Robert D. 77
Reddy, S. M. 233, 238
Robbins, W. E. 567
Root, William L. 1039
Rosenbrock, H. H. 1, 574
Rutman, R. 450
Saeks, R. 942
Sarma, V. V. S. 749
Sawin, Dwight H., III 224
Schmidt, Douglas C. 106
Sedra, A. S. 547
Shamash, Y. 450
Sheih, L. S. 957
Siljak, D. D. 459, 608
Singh, H. 967
Singhal, K. 713
Sinha, A. S. C. 485
Siret, J. M. 809
Small, Henry G. 512
Snyder, Wesley E. 863
Sohal, J. S. 967
Steiglitz, K. 120
Strintzis, Michael G. 185, 207

Subrahmanyam, M. B. 932
Summit, Roger K. 522
Sun, K. G. 897
Tan, Harry H. 740
Thomas, A. Thampy 658
Thomas, G. 749
Thomas, John B. 1007, 1029
Tsui, Frank 268
Underwood, Stephen A. 862
Vincentelli, A. Sangiovanni 686
Viswanadham, N. 351
Vlach, J. 713
Vranesic, Z. G. 547
Wadhwani, R. M. 730
Wang, Paul P. 288
Waltz, D. L. 875
Weissenberger, Stein 87
Wells, Francis M. 720
White, Lee J. 398
Willsky, Alan S. 849
Wise, G. L. 1029
Wolfe, Philip 8, 35
Wonham, W. M. 457
Wooley, B. A. 535, 542
Yau, Stephen S. 307
Yoshida, Nobuyoshi 705
Zobrist, George W. 695